APPLICATIONS OF ARTIFICIAL NEURAL NETWORK CONCEPTS IN TRAFFIC ENGINEERING : PATTERN RECOGNITION OF TRAFFIC ELEMENTS

A THESIS SUBMITTED

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF TECHNOLOGY

by

VIJAY PRATAP SHARMA

to the

DEPARTMENT OF CIVIL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

MAY 1993

10/5/93

P.N.

CERTIFICATE

It is certified that the work contained in the thesis entitled, "APPLICATIONS OF ARTIFICIAL NEURAL NETWORK CONCEPTS IN TRAFFIC ENGINEERING: PATTERN RECOGNITION OF TRAFFIC ELEMENTS" by Shri Vijay Pratap Sharma, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

10th May, 1993

(DR. S. P. Palaniswamy)
Professor
Department of Civil Engineering
I.I.T Kanpur

ABSTRACT

A convenient and easy method of traffic interaction study, for the determination of various traffic parameters, is the analysis of video recorded data. A special Video Instrumentaion System mounted on Maruti van (as test vehicle) has been used to record 3D scenes of traffic movement, on various roads, onto video cassettes. It is in this form that the data is available and the analysis of this recorded data is being done manually using hardware, and software.

A detailed study of Artificial Neural Networks and Image Processing has been carried out and a strategy chalked out to make an attempt to replace the process of manual data processing by automatized image processing because artifical neural networks, in association with a good vision system, may prove to be excellent for 3D visual scene analysis and may reduce the time of analysis considerably.

As a first step towards this automatization, an attempt has been made for vehicle shape recognition using Artficial Neural Networks, which may be used for vehicle dynamics later on. A software on artificial neural network has been developed which has been trained by a number of patterns

(corresponding to different views of a vehicle). These patterns have to be in the language of computers i.e., in the form of numbers (nxn matrices) which are obtained by analog to digital conversion of the 2D picture images (of the 3 D scenes) of different views of the vehicle. The different views of the vehicle have been obtained by video recording using the video instrumentation system fixed in a Maruti van.

The artificial neural network is a means to non programmed adaptive information processing. The network is trained by some known patterns so that it may develop itself into such a state so as to recognize similar patterns. Supervised training has been used in the software wherein the outputs corresponding to the input patterns are known. The implementation of the software needs consideration of several aspects to avoid practical problems. Even a 8x8 input matrix (i.e., 64 inputs) require a large memory and a 512x512 matrix may require a memory of 100 MB or higher. Moreover a large CPU time is required to bring down the error to a value within limits.

The error after training has been found to be sufficiently low and is a healthy sign towards the future use of artificial neural networks for image processing. The training of the network by these patterns consumes quite a large CPU time, requires a large number of iterations and has

a large memory requirement. The higher is the order of the matrices (patterns), the more is the memory requirement and this imposes certain restrictions on the order of matrices used thus increasing the error which might, otherwise, have been even lesser.

Dedicated To

MY BELOVED PARENTS

ACKNOWLEDGEMENTS

To the utmost depth of my heart, I can't visualise the epitomizing of my thesis work without the help of some of my nears and dears.

First and the foremost, I would like to express my deep sense of gratitude and appreciation to *Dr. S. P. Palaniswamy* for his able and dynamic guidance and his relentless efforts which were instrumental in the completion of this work.

Thanks are also due to the faculties of Civil Engg. Deptt. under whose able guidance I could gain a knowledge of this field.

A special word of appreciation is due to Dr. P. K. Kalra and Mr. S. K. Joshi for their enthusiastic support at all stages during the course of these investigations.

The cooperation extended by Manu, Rajesh, Bhuvnesh, Kaalga, Shailaja, Mrs. Valsala, Bhim and Gautam in bringing this work to black and white is gratefully acknowledged.

Last, but not the least, I am highly indebted to my parents, Sri. I. P. Sharma and Mrs. R. Sharma, my brother Lieut. A. P. Sharma and my sister Dr. Archana Sharma for encouraging me to strive for excellence and without whose blessings this work would never have been possible.

VIJAY PRATAP SHARMA

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 GENERAL

With fast increasing traffic volumes in the wake of all round development, the need for upgrading and enhancing the transportation network makes itself strongly felt making substantial progress on limited national resourses. A very salutory effect of this constraint of resourses has been the focussing of attention on the need for accurate techno economic analysis of various design alternatives so as to attain maximum returns, for example from the highway sector, in terms of minimum transportation cost. For fulfiling the need for accurate and quick analysis, it is quite natural that researchers began to search for automatized methods and this is how Artificial Intelligence (AI) and Artificial Neural Networks (ANNs) began to be looked upon as effective tools for this automatized analysis in the field of Transportation Systems also. To be more precise, presently, our main intention is the use of AI and ANN in the field of Image Processing for determination of various parameters, in an automatized manner so that this may help us in the design

of automatized intersection controls, automatized accident warning systems etc. This will also help fulfill the urgent need for traffic simulation modelling for Indian conditions. In fact a model developed at National Road and Traffic Research Institute of Sweden was considered to be the most suitable for modification to suit the Indian conditions. There was an agreement between the Government of India (Ministry of Shipping and Transport) and Swedish International Development Agency according to which VTI and CRRI were to carry out this study and, boastfully on the Indian side I.I.T. Kanpur was assigned the analytical aspect and that is what Image processing is being used for.

Now, before moving into the details of Image processsing and Pattern Recognition phenomena, we must have an insight into the fields of AI and ANN so that we are able to make out how they can be used for automatised image processing and parameter determination.

1.2 ARTIFICIAL INTELLIGENCE

As the name signifies and what is a layman's understanding of this term, it is something that is a replica of natural intelligence. It has been a goal of science and engineering to develop intelligent machines for many decades. These machines were envisioned to perform all tedious and

cumbersome tasks so that we may enjoy a more fruitful and enriched life. AI is a branch of computer science concerned with the study and creation of computer systems that exhibit some form of intelligence, systems that learn new concepts and tasks, systems that can reason and draw useful conclusions, systems that can understand a natural language or perceive or comprehend a visual scene, and systems that perform other types of feats that *require human type of intelligence.*

Intelligence is not only the ability to exercise thought and reason as defined in the dictionaries. It, in fact, embodies all of the knowledge and feats, both conscious and unconscious which we have acquired through study and experience. It is the integrated sum of all those feats which give us the ability to recogize a face not seen for 30 years or more or gives us the ability to send rockets to the moon. It is in fact those capabilities which set *Homo sapiens* apart from other living beings

Can we ever expect to build systems which have these chracteristics? Yes, of course, is the answer. Systems have already been developed to perform many types of intelligent tasks and expectations are high for near future deveopment of even more impressive ones.Now we have systems which can:

(i) learn from examples, from being told, from past related

experiences.

(ii) solve complex problems in scheduling, optimization, planning of miliatry strategies, digonosing deseases.

(iii) *see well enough to "recognize" objects from photographs, video cameras and other censors.*

(iv) understand large parts of natural language.

But we still have not been able to produce co-ordinated autonomous systems which possess the abilities of even a three year old child which include ability to :

* recognize and remember diverse objects in a scene.
* learn new sounds and associate them with objects and concepts
* adapt readily to many diverse situation

The above mentioned inabilities are in fact the challenges facing AI researchers.

Precisely, in AI the goal is to develop working computer systems that are truly capable of performing tasks that require high levels of intelligence. A better understanding of AI is gained by looking at the component areas of study that make up the whole. These are:

* Robotics
* Memory organization
* Knowledge representation
* Storage and recall

* Learning models

* Inference techniques

* Commonsense reasoning

* Understanding natural language

* Pattern recognition

* Machine vision methods

* Search and matching

* Speech recognition and synthesis

Importance of AI:

AI may be one of the most important developments of this century and countries leading in this field will emerge as the dominant economic powers of the world. Japanese were the first to demonstrate their commitment to this field when they launched a very ambitious program in AI in October 1981, called as *Fifth Generation* which calls for implementation of a 10 year plan to develop intelligent supercomputers, with a combined budget of about 10 bilion dollars. If they succeed their position as a leading economic power is assured.

Reasearches in the field of AI are also in progress in Britain, France, U.S. and one thing is clear that the future of a country is closely tied to the commitment it is willing to make in funding research programs in AI.

Fields closely related to AI are Mehanical Engineering, Electrical Engineering, Linguistics, Psychology, Cognitive

Sciences, Philosphy, and Robotics.

Applications of AI have been proven in Civil Engineering, Defense, Chemistry, Biology, Banking, Economics, Manufacturing, Law , Medicine, Aerospace etc.

1.3 ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANNs) are a part of AI, rather complementary to it. Imagine a computer that learns wherein information is fed into it alongwith examples of the conclusions it should be reaching or feedback on how it is doing or the machine may even be left to its own devices. The information processing system of the machine called the *Artificial Neural Network involves a non algorithmic approach* wherein the computer simply runs through the material again and again making myriads of mistakes but learning from them untill finally it gets itself into proper shape to carry out the task successfully. Such behaviour of the ANNs are quite human and their design is inspired by the stucture of the human brain and working of the brain cells, the *neurons* as they are called and hence the name.

<u>Neurocomputing</u>:

It is the Engineering discipline concerned with *non programmed adaptive information processing systems: the*

neural networks that develop associations between objects in response to their environment. It is a fundamentally new and different information processing paradigm: the first alternative to algorithmic programming. Its application may reduce development costs and time often by an order of magnitude. It does not however replace algorithmic programming, it being in a state of infancy and applicable to only certain types of problems. It is suspected that neurocomputing and algorithmic programming may be conceptualy incompatible.

In fact, ANN are biologically inspired i.e., they are composed of elements that perform in a manner that is analogous to the most elementary function of the biological neuron, the brain cell. These elements are then organised in a way that may (or may not be) related to the anatomy of the brain. Despite this superficial resemblence, ANNs exhibit a surprising number of the brain characterstics. For example, they learn from experience, generalise from previous examples to new ones and abstract essential characterstics from inputs containing irrelevant data. Inspite of all this, it can not be suggested that ANN will soon duplicate the functions of human brain. The actual intelligence exhibited by the most sophisticated ANN is below the level of a tapeworm. This reality should be kept in mind to check over enthusiasm.

However it is equally incorrect to ignore the surprisingly brainlike performance of certain ANNs.

1.4 PROBLEM STATEMENT AND OBJECT OF STUDY

The manual image processing which is currently being done is a tedious process. Moreover, there is every possibility of manual errors effecting the results. Hence it has become rather imperative to automatize this process of 3D scene analysis. As an era of ANNs has again begun after an eclipse of 10 years and this complement of AI is being found useful for tackling such problems, it has been planned to make efforts to use ANNs for our purpose. But, it is still a very new field wherein trial and error is going on and whatever theory has been propagated is still, probably, in the form of hypothesis and not a law. The results achieved also range from a meagre 5% to as high as 98%.

The object of the present study is two fold. At the first instance an indepth knowledge of the field of ANNs has been gained which comprises of the fundamentals of ANNs, types of ANNs, types of training of ANNs, the algorithm involved in training and the working of ANN. Thus we could know the present *state of this art.* Secondly an extensive study revealed how ANNs can be applied to the present field: Image processing in Traffic Engineering. An effort has been

made to apply the knowledge to a practical problem of shape recognition of vehicle. The 3D views of vehicle at different angles were converted into 2D picture frames and these were digitised as per a definite scheme as is used in the case of alphabet recognition. A software on ANN using Backpropagation algorithm has been developed and an attempt has been made to train the ANN with these digitised images of vehicles so that the network gets trained to recognize any similar pattern (digitized image) which it may be exposed to.

1.5 ORGANISATION OF THESIS

The thesis has been organised under the following heads:

1. An introduction to AI and ANN, problem statement, object of study and organisation of thesis is contained in Chapter 1.
2. Chapter 2 contains an extensive literture review including the work done in the international arena in recent past on ANNS specially their applications to Civil Engg.
3. Chapter 3 contains the fundamentals, types, details of training and working of ANNs together with a description of Backpropagation Algorithm.
4. The whole process of image processing, description of 3D computer vision systems, analog to digital conversion of visual scenes, and procesing of quantised data are the topics

visual scenes, and procesing of quantised data are the topics which have been dealt with in Chapter 4.

5. A description of the manual processing (including instrument setup) and the proposed automatised processing besides a desription of the software developed for the purpose form the contents of Chapter 5.

6. The summary, conclusions, and scope for future work are the topics that have been included in the concluding chapter, the Chapter 6.

CHAPTER 2

LITERATURE REVIEW

2.1 HISTORICAL PERSPECTIVE

The improved understanding of the functioning of the neuron and the pattern of its interconnections has allowed researchers to produce mathematical models to test their theories. Two mutually reinforcing objectives of neural modelling were defined and remain today :

1. To understand the physiological and psychological functioning of the human neural system.
2. To produce ANNs that perform brain like functions.

Models of human learning were developed of which one, that has proven to be the most fruitful, was that of D.O.Hebb in 1949 (Wasserman, 1989). He proposed a learning law that became the starting point for ANN training algorithms.

Early successes produced a burst of activity and optimism and networks consisting of single layers of artificial neurons were developed called PERCEPTRONS which were applied to diverse fields like weather prediction, electrocardiogram analysis and artificial vision.

But Marving Minsky's researches led to the publication of his book *Perceptrons* in which he and Seymore Papert proved

including the functions performed by a simple exclusive OR gate (Wasserman, 1989). Minsky was not optimistic about progress even. This discouraged most of the researchers. Yet, a few dedicated scientists such as Teuvo Kohonen, Grossberg, Anderson etc continued their efforts. Gradually a theoretical foundation emerged upon which the more powerful multilayered neural networks of today have been constructed.

ANNs today:

There have been many impressive demonstrations of ANN capabilities, a few of which using one bacpropagation Algorithm are :

S.No	Functions Performed	Inventor
1.	Conversion of text to speech	Seijnowsky and Rosenberg(1987)
2.	Recognition of handwritten characters	Burr, 1987
3.	Image Compression	Ottrel,Munro and Zipser 1987

Many other algorithms have been developed and been used in other types of networks.

2.2 TRAINING NEURAL NETWORKS FOR TRAFFIC CONGESTION

(For specific terms used chapter 3 may be referred)

A major problem for all automatic control systems, both urban and rural, is deciding whether a particular road link is congested. This is because "congested" is a very subjective function and depends on the context in which the link is situated. A second difficulty is that the value of a single parameter such as vehicle flow or occupancy is not usually sufficient to diagnose congestion. Hence a variety of factors are used by an experienced operator viewing a link by a CCTV to decide that a particular link is congested. The unfortunate part is that it is not easy to express this type of decision making algorithmically. Another problem is that congestion on a link can't be removed without affecting other links which necessitates the consideration of a sub area of several links. A likely solution to this problem is the Artificial Neural Network approach as the network itself would develop the required relationships between data for different links.

EXPERIMENTAL APPROACH: To explore the possible ways of approaching this, problem data were provided by the University of Nottingham, collected via an on line computer link to the SCOOT traffic control system of Leicester. The

Figure21(a)RMS Error (Single Parameter)

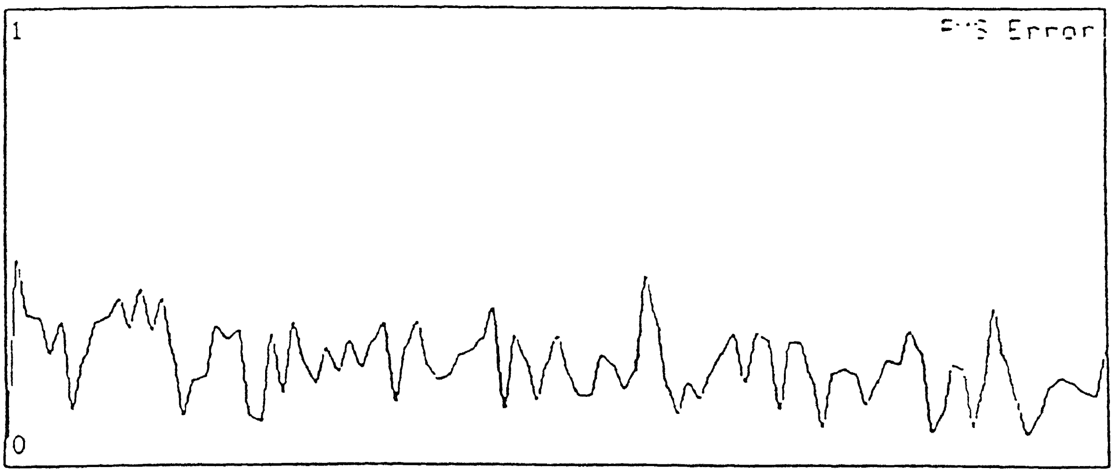


Figure21(b)RMS Error Curve (Two Parameters)

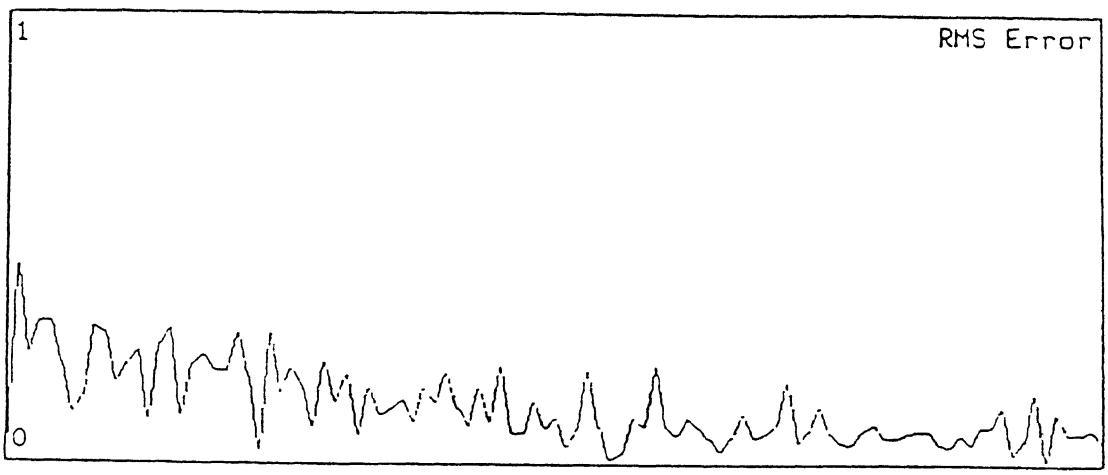
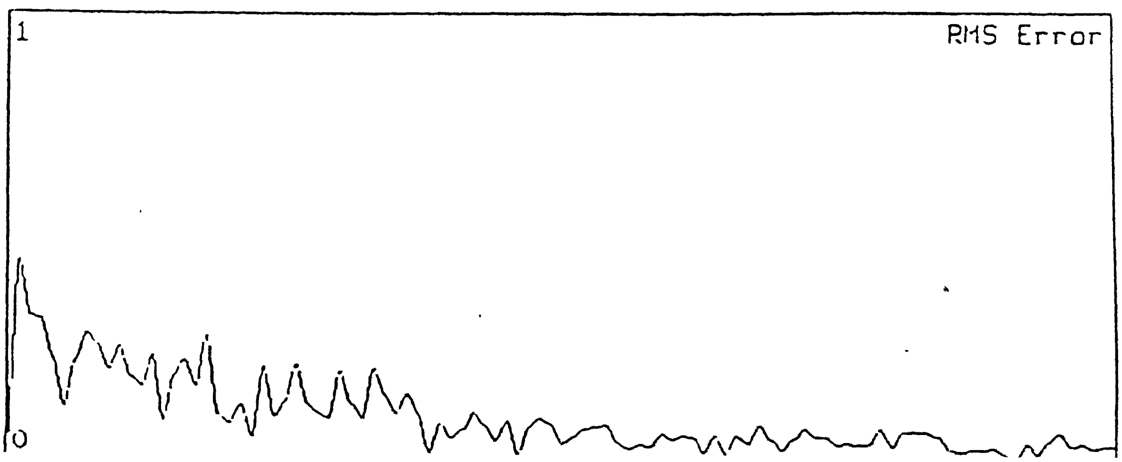


Figure21(c)RMS Error curve (Three Parameters)



data consisted of values recorded over 20 days every 5 minutes, of 3 different parameters over 40 links: vehicle flow, queue length, and percentage of total free flow capacity used. These parameters were checked against an on street survey carried out as part of another project. An expert familiar with the road system, then, provided a list of time periods when a small sub area was considered to be congested after thoroughly going through the data. It was then analyzed whether a model could be built which carried out the same diagnosis.

The approach taken was to train a back propagation neural network using the data set collected. The inputs consisted of the current congestion parameters from 20 neighbouring detector sites, normalised to between 0 and 1. The output was a binary variable indicating whether the sub area was considered to be congested or not.

Three different classes of experiments were conducted wherein the training set numbered about 400 examples for each class (for details of training refer to Chapter 3). The training was halted manually when either of the following two conditions occured :

(i) The RMS error had decayed to nearly zero and variations were no longer considered significant.

(ii) The RMS error was oscillating randomly within certain

ranges and further convergence appeared unlikely. This condition depends on intuition and is easily recognizable by experience.

One of these criteria was, typically, fulfilled after about 12,000 iterations and hence graphs of RMS error were presented for 16,000 iterations.

Class I :

Three neural networks were trained using each congestion parameter in turn but in each case the training was halted after little or no convergence was observed (Fig.2.2a).

Class II :

Three more neural networks were trained, covering all possible combinations of two parameters. Some convergence was noted for all the three networks, although when the training was stopped the RMS error has not reached zero, and quite considerable oscillation was still apparent (Fig. 2.2b).

Class III:

A neural network was, finally, trained with all the three parameters and the convergence was found to be quite good with little final instability. (Fig. 2.2c).

CONCLUSIONS: The data set being not very large care has to be taken while interpreting this set of results. In an ideal case the neural network must be tested after training by presenting a further set of data which may not be possible

due to limited data. However, the fact that convergence occured is a clear indication that an ANN approach is likely to be successful and further research must follow. The fact that convergence occured with all the three parameters taken together implies that congestion can be better diagnosed by observation of several parameters and that ANNs seem to be a good solution to the problem of interpreting large amounts of data which are interrelated but for which no straight forward algorithm can be found.

2.3 AN OVERVIEW OF RECENT RESEARCH IN THE FIELD OF ANN

The following works have been done in international arena in the recent past:

1. A paper on road traffic monitoring using one TRIP 2 system has been presented. This paper contains a brief review of present image processing systems used for traffic monitoring, including a discussion of the disadvantages of such systems. The vehicle detection algorithm relies upon the ability of TRIP 2 system to learn from example and discriminate between complex patterns within video images of traffic scenes. During site trials of the system it was possible to detect 99% of the vehicles and individual vehicle speeds to an accuracy of between plus or minus 8% & 17% respectively

(Dickension, K.W. and Wan, C. L.).

2. A Self Organising Traffic Controll System has been developed using Neural Network Model (Nakatsuji, T. and Kaku, T., 1991).

3. Neural Networks have also been used for automated vehicle dispatching. An alternative Neural Network Model was proposed as a sub symbolic and empirical alternative for modelling the decision process of expert dispatchers (Potwin, J.Y., Shen, Y., and Rousseau, J.M.).

4. An Intelligent System for Automated Pavement Evaluation has been developed. This research is directed towards an innovative, noncontact, intelligent, nondestructive evaluation (INDE) system, using a novel AI based approach that would integrate 3 AI technologies viz. Computer vision, Neural networks, and Knowledge based expert system wherein multilayer perceptron and backpropagation learning rule have been used (Ritchie, S.G., Kaseko, M., and Bavarian, B., 1991)

5. A Neural Network Model for Freeway Incidence detection has been developed. This paper presents the initial results of an exploratory study investigating the application of Neural pNetwork Models from the field of AI to the automated detection of non recurring congestion on urban freeways. The results are encouraging (Cheu, R.L., Ritchie, S.G., Recker, W.W. and, Bavarian, B.)

6. A paper on pavement image processing has been presented wherein the potential for employing neural network model in the pavement image interpretation has been discussed and preliminary results presented (Kaseko, M.S., and Ritchie, S.G.)

7. A self organising traffic control system using neural network models has been developed.(Nakasuji, T., and Kaku, T., 1991).

8. A modular neural network has been developed for recognition of car registration plates (Margarita, G., 1990).

9.The drivers' behaviours using a driving simulator have been studied (Takubo, N., 1991).

10. Neural Network has been used in the development of an autonomous land vehicle (Pomerleau, D.A., 1989)

CHAPTER 3

THEORY

3.1 FUNDAMENTALS OF ANNs

As already mentioned, ANNs are biologically inspired, that is, when considering network configuration and algorithms, the researchers usually keep in mind the organization of the brain. Knowledge about the brain's overall operation is so limited that there is little to guide those who would emulate it. Hence, network designs must go beyond current biological knowledge seeking structures that perform useful functions. Despite this the artificial neural networks continue to evoke comparisons with the brain. These functions are often reminiscent of human cognition.

The human nervous system is built of cells called neurons and is a highly complex structure. An estimated 10^{11} neurons participate in perhaps 10^{15} interconnections over transmission paths that may range for a metre or more. Each neuron shares many characteristics with the other cells in the body but has unique capabilities to receive, process and transmit electrochemical signals over the pathways that comprise the brain's communication system. A study of the

structure of the neuron will clarify the working.

Before going into details, it is necessary to get acquainted with the following terms that shall be used frequently :

NEURON: It is the basic building block of a Neural Network, also known as a neuron or processing element.

CONNECTION: a signal transmission pathway between processing elements, corresponding to axons and synapses in a human brain.

LAYERS: In theory any topological arrangement should work but for ease in analysis and visualisation it is usual to arrange the nodes in layers with all nodes in adjacent layers connected to each other.

WEIGHTS: an adaptive coefficient associated with a single input connection. It determines the intensity of the connection.

PROCESSING ELEMENT: an artifical neuron in a neural network consists of a small amount of local memory and processing power and hence the name.

LEARNING LAW: an equation that modifies some or all of the adaptive coefficients (weights) in a processing element's local memory in response to input signals and the values supplied by the transfer function.

TRANSFER FUNCTION: a mathematical formula that, amongst other things, determines a processing element's output signal as a function of the most recent input signals and the weights in local memory.

AXON: the connection emerging from the cell body of a typical biological neuron.

DENDRITES: finer sub connections emerging from axon.

Structure of a pair of typical biological neuron (3.1 a) :

The figure 3.1(a) shows the structure of a pair of typical biological neurons. Dendrites extend from the cell body to other neurons where they receive signals at a connection point called synapses. On the receiving side of synapses these signals are conducted to the cell body where they are summed, some inputs ending to excite the cell whereas others tending to inhibit its firing. When the cumulative excitation in the cell body exceeds a threshold the cell fires, sending a signal down the axon to other neurons. Although this basic fundamental outline has many complexities and exceptions yet most of the ANNs model only these simple characteristics.

The artificial neuron (Fig.1 b.) :

This was designed to mimic the first order characteristics of the biological neuron. The working is :

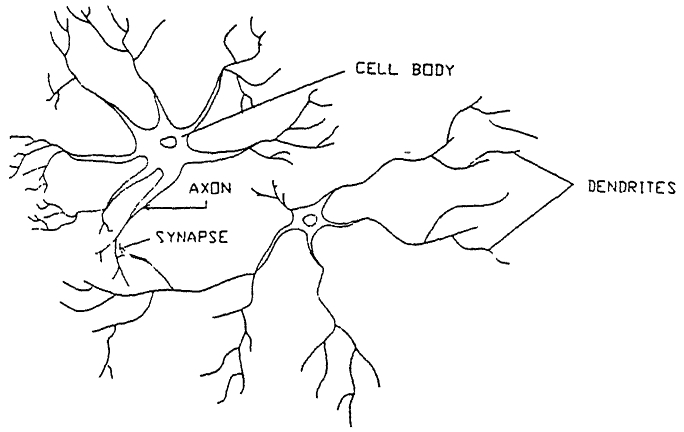


FIG.3·1(a) Biological Neuron

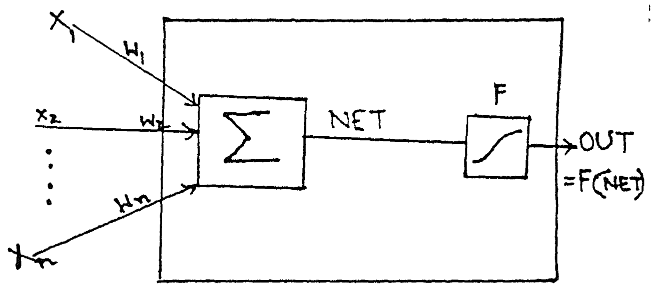


FIG 3·1(b) Artificial Neuron

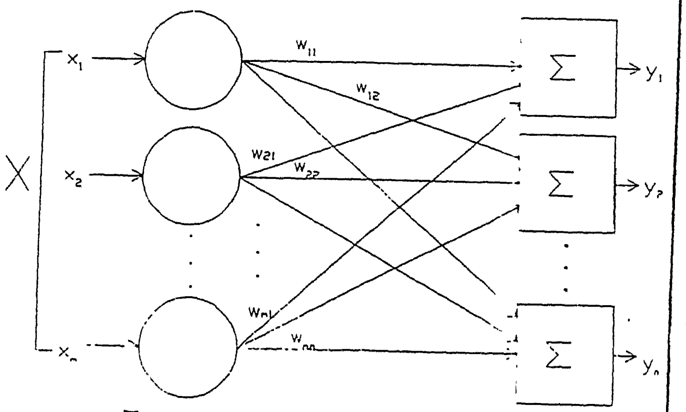


FIG 3·1(C) Single-Layer Neural Network

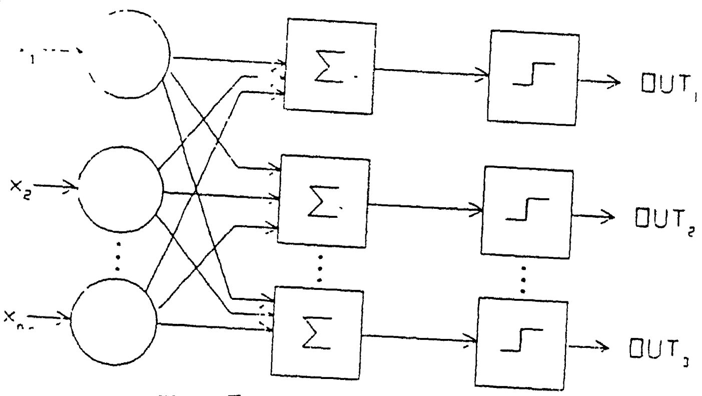


Figure 3·1(d) Multioutput Perceptron

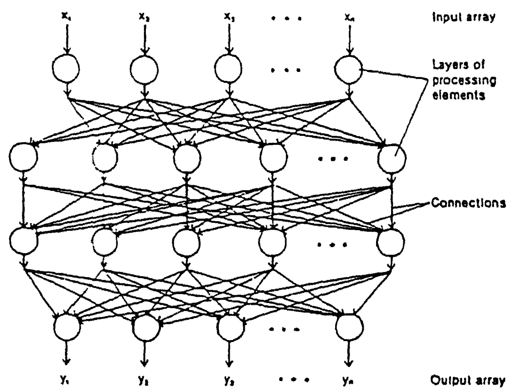


Figure 3.1(e) MULTILYERED NEURAL NETWORK

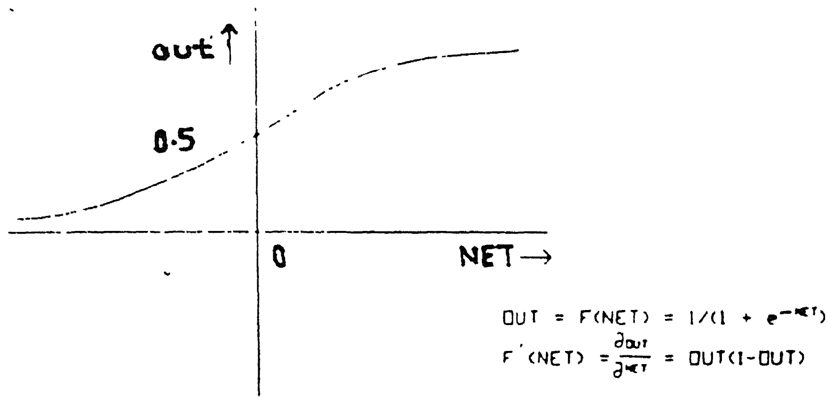


Figure 3-2(a) Sigmoidal Activation Function

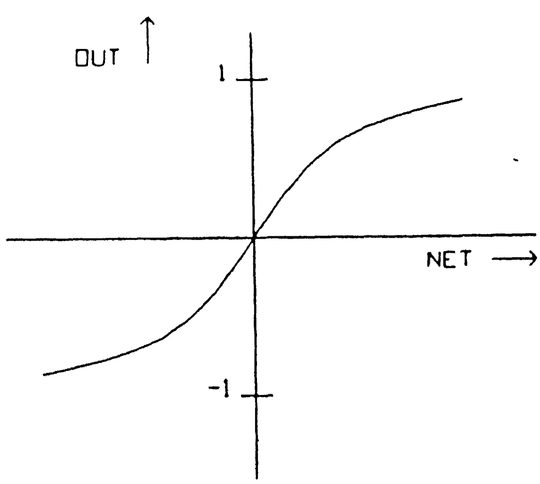


Figure 3-2 b. Hyperbolic Tangent Function

* A set of inputs, each representing the output of another neuron, are applied.
* Each input is multiplied by a corresponding weight analogous to a synaptic strength.
* All the weighted inputs are then summed to determine the activation level of the neuron.

Although there are lot many network paradigms but all are based upon this configuration. As per the figure, the output is,

$$NET = \overline{XW} \tag{3.1}$$

where,

X=vector representing the set of inputs

W=vector representing corresponding weights (synaptic strengths)

Activation Function :

The NET signal is usually further processed by an activation function F to produce the neuron's output signal OUT. This may be in any of the following form :

(a) a simple linear function,

$$OUT = K\ (\ NET\ ) \tag{3.2}$$

where K is a constant, a threshold function

OUT= 1 if NET >T

OUT= 0 otherwise,

T is a constant threshold value or a function which more

accurately simulates the nonlinear transfer characteristics of a biological neuron.

In the Fig. 3.1 (b) the block labled F accepts the NET output and produces the signal OUT. If the F processing block compresses the range of NET so that OUT never exceeds some low limits regardless of the value of NET, then F is called a squashing function. The squashing function is often chosen to be a logistic or sigmoid (meaning S shaped) function.

Mathematically, it can be expressed as,

$$F(x) = 1/(1+e^{-x}) \tag{3.3}$$

Hence,

$$OUT = 1 /(1 + e^{-NET}) \tag{3.4}$$

The central high gain region of the logistic function solves the problem of processing small signals while its regions of decreasing gain at the positive and negative extremes are appropriate for large excitations.

(c) Sometimes hyperbolic function OUT = tanh(x) is also used (fig 3.2b). Like the logistic function, this is also S shaped but is symmetrical about the origin resulting in OUT having the value zero when NET is zero. Unlike the logistic function the hyperbolic one has a bipolar value for OUT which is beneficial in certain networks.

Single layered ANNs (Fig. 3.1 c)

Although a single neuron can perform certain simple pattern detection functions, the power of neural computation comes from connecting neurons into networks. The simplest network is a group of neurons arranged in a layer as shown in the Fig 3.1c. The circular nodes only serve to distribute the inputs, they perform no computations and hence will not be considered to constitute a layer. The set of inputs $\underline{X}$ has each of its elements connected to each artificial neuron through a separate weight. Early ANNs were no more complex than this.

Therefore calculating outputs N from a layer is simple matrix multiplication.

$$N = X W \tag{3.5}$$

If this sum is greater than a predetermined threshold, the output is one otherwise it is zero. These systems and their many variations collectively have been called *Perceptrons* (Fig. 3.1 d). Despite the limitations of Perceptrons, they are a logical starting point for a study of ANN.

REPRESENTATION : It refers to the ability of a perceptron to simulate a specified function. A single layered perceptron is seriously limited in its representational ability. There are many simple machines that the perceptron can't represent no

matter how the weights are adjusted. One of Minsky's more discouraging results shows that a single layered perceptron can't simulate a simple exclusive OR function.

Multilayered ANNs (Fig. 3.1 e)

Greater computational capabilities are offered by larger more complex networks. Multilayered networks may be formed by simply cascading a group of single layers, the output of one layer providing input to the subsequent layer. Calculation of the output of the layer is,

$$N = (X \overline{W}_1) \overline{W}_2$$

$$= X (\overline{W}_1 \overline{W}_2) \tag{3.6}$$

where W_2 represents the second weight matrix. This shows that a two layered linear network is exactly equivalent to a single layered one having weight matrix equal to the product of two weight matrices.

Recurrent Networks

More general networks that do contain feedback connections are said to be Recurrent Networks.

3.2 TRAINING AND WORKING OF ANNs

Out of all the interesting characteristics of ANNs the most important and eye catching is their ability to learn.

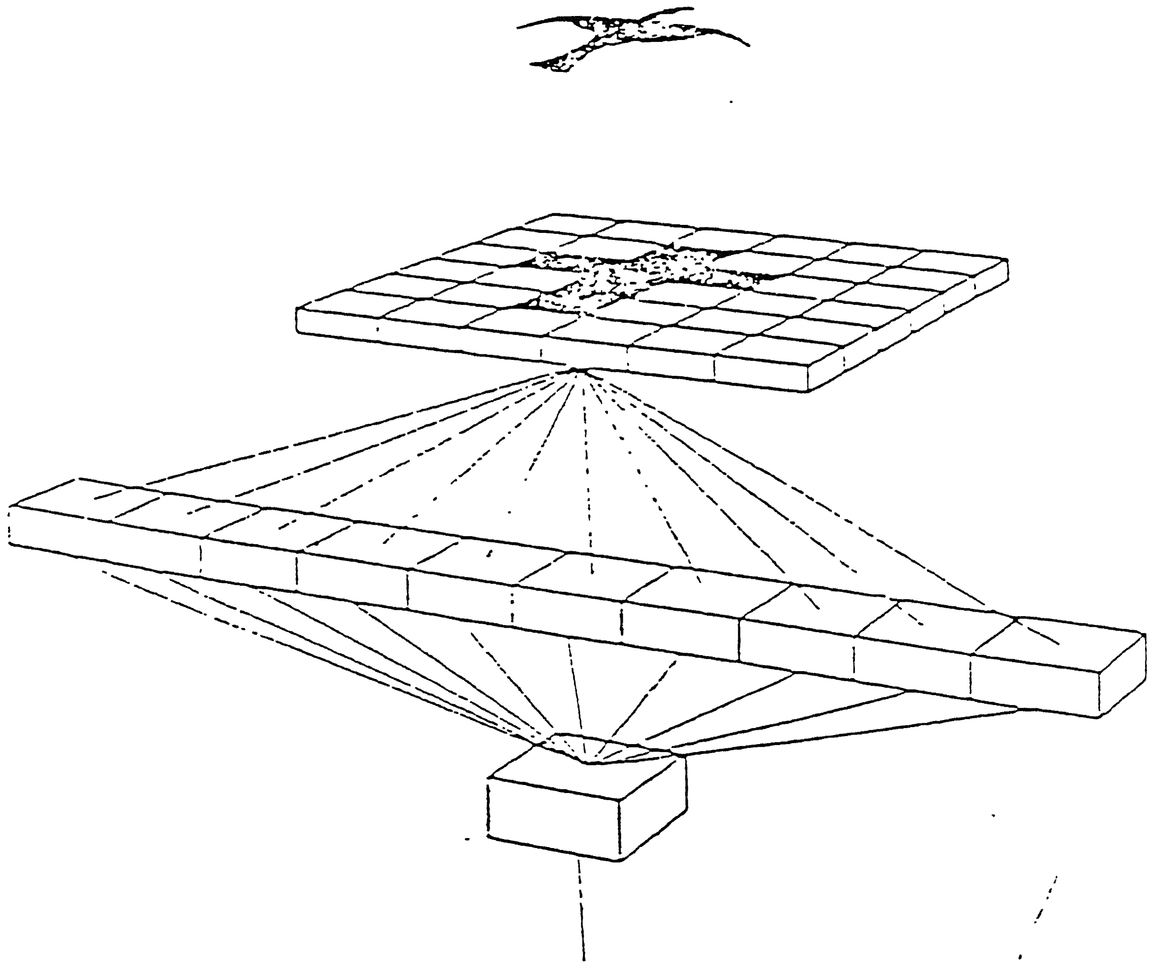


FIG. 3.3 NEURAL NETWORK WORKING

Their training shows so many parallels to the intellectual development of human beings that it may seem that a fundamental understanding of the process has been achieved (Fig. 3.3 gives a general idea of the working of ANNs wherein a simple artificial neural network consisting of a retina 6x6 receptor cells, 10 hidden cells, and 1 output cell has been shown. Only those connections originating from one of the receptor cells have been illustrated for clartiy).

Learning in ANNs is limited and many difficult problems remain to be solved before it can be determined if we are even on the right track.

Objective of Training :

A network is trained so that application of a set of inputs produces the desired or at least consistent set of outputs. Each such input (or output) set is referred to as a vector. Training is accomplished by sequentially applying input vectors while adjusting network weights according to a predetermined procedure. During training the network weights gradually converge to values such that each input vector produces the desired output vector.

Types of Training :

Training can be categorised into :

a) Supervised training : This requires the pairing of each input vector with a target vector representing the desired

output; together these are called a Training Pair.

Usually a network is trained over a number of such training pairs. An input vector is applied, the output of the vector is calculated and compared to the corresponding target vector. The difference (error) is fed back through the network and weights are changed according to an algorithm (Backpropagation being the most common) that tends to minimize the error. The vectors of the training set are applied sequentially, the errors calculated and the weights adjusted for each vector until the error for the entire training set is at an acceptably low value.

b) Unsupervised Learning: Inspite of many successes, supervised training has been criticized it being difficult to conceive of a training mechanism in brain that compares desired and actual outputs feeding processed corrections back through the network. Several questions remained unanswered :

If this were the brain's mechanism, where do the desired outputs come from?

How could the brain of an infant accomplish the self organization that has been proven to exist in early development?

Thus unsupervised learning is a far more plausible model of learning in the biological system. Developed by Kohonen (1984) and many others, it requires no target vectors for the

outputs and hence no comparisons to predetermined ideal responses. The training set consists solely of input vectors and the training algorithm modifies network weights to produce output vectors that are consistent. The training process therefore groups similar vectors into classes.

Training Algorithm :

Most of today's training algorithms have evolved from the concepts of D.O Hebb (1961). He proposed a model for unsupervised learning in which the synaptic strength (weight) was increased if both, the source and destination neurons, were activated. In this way often used paths in a network are strengthened and the phenomena of habit and learning through repetition are explained.

An ANN using Hebbian learning will increase its network weights according to the products of the excitation levels of the source and destination neurons.

However more effective learning algorithm including those for supervised learning have been developed.

The Delta rule

This is an important generalisation of the perception training algorithm which extends this technique to continuous inputs and outputs. The perception training algorithm may be generalised by introducing a term δ which is the difference between the target output T and the actual output A.

$$\delta = (T - A) \quad (3.7)$$

For the correction associated with the i^{th} input

$$\Delta_i = \eta \delta x_i \quad (3.8)$$

$$W_i(n + 1) = W_i(n) + \Delta_i \quad (3.9)$$

where η = learning rate coefficient,

$W_i(n + 1)$ = the value if weight i after adjustment,

$W_i(n)$ = the value of weight i before adjustment.

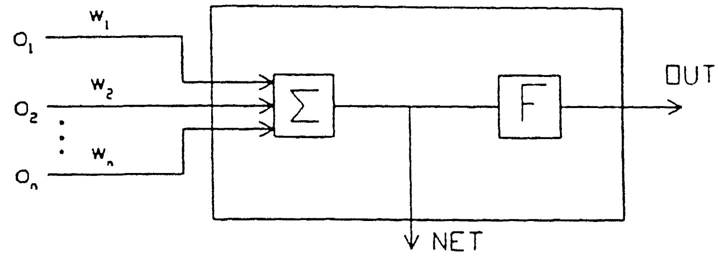
The delta rule modifies weights appropriately for target and actual outputs of either polarity and for both continuous and binary inputs and outputs.

3.3 BACKPROPAGATION ALGORITHM

For supervised training of multilayer artificial neural network, the algorithm which proved to be a boon is the back-propagation algorithm which shall be discussed in detail here.

It is a systematic method for training multilayer artificial neural networks. Despite its limitations it has dramatically expanded the range of problem to which artificial neural networks can be applied.

As discussed earlier, the neuron as shown (Fig.3.4a) is used as the fundamental building block for back propagation algorithm.



$$NET = o_1 w_1 + o_2 w_2 + \dots + o_n w_n = \sum_{i=1}^{n} o_i w_i$$

$$OUT = F(NET)$$

Figure 3.4(a) Artificial Neuron with Activation Function

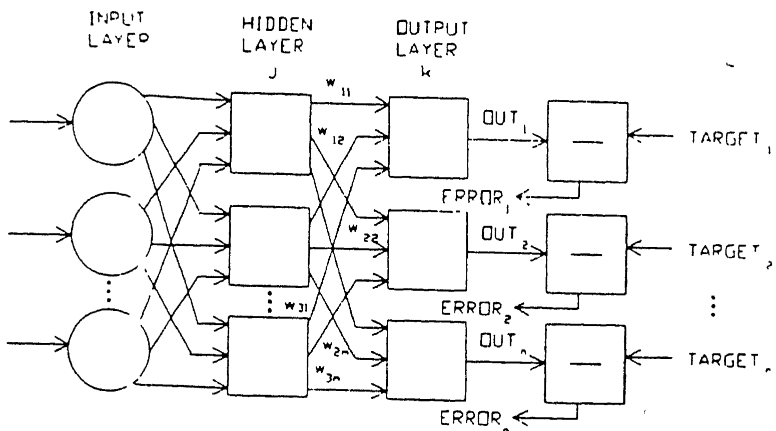


Figure 3.4(b) Two-Layer Backpropagation Network

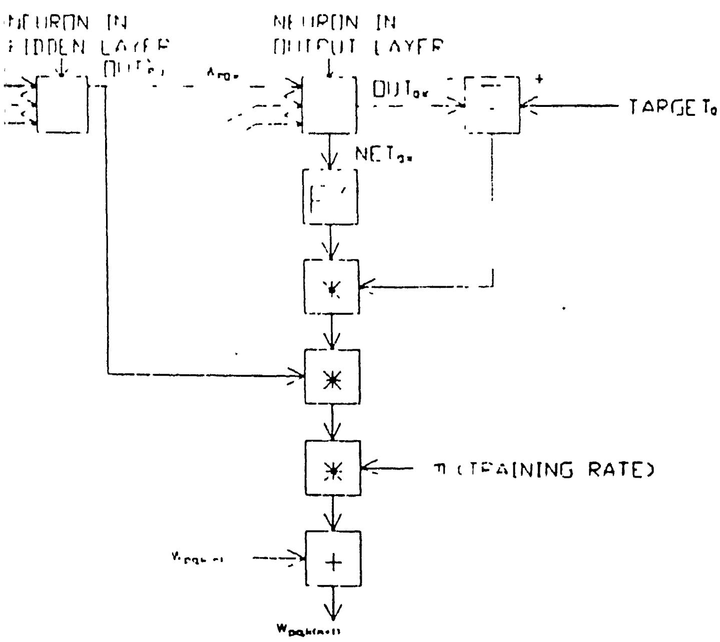


Figure 3.4(c) Training a Weight in the Output Layer

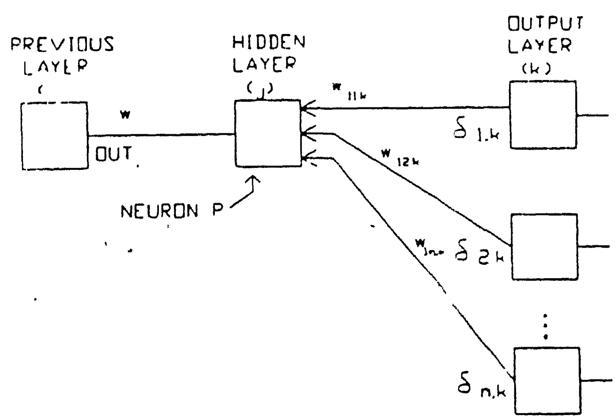


Figure 3.4(d) Training a Weight in a Hidden Layer

$$NET = O_1W_1 + O_2W_2 + \dots + O_nW_n = \sum_{i=1} O_iW_i \quad (3.10)$$

$$OUT = F(NET) \quad (3.11)$$

Activation function usually used for backpropagation is

$$OUT = F(NET) = 1/(1 + e^{-NET}) \quad (3.12)$$

$$F'(NET) = \partial\ OUT/\partial\ NET = OUT(1 - OUT) \quad (3.13)$$

The sygmoid often called logistic or squashing function comprises the range of NET so that OUT lies between 0 and 1. The squashing function produces non linearity resulting which multilayer networks have greater representational power. Some other functions may also be used for backpropagation provided they are differentiable every where.

Back propagation can be applied to networks with any number of layers, however it can easily be understood by demonstration on a network with two layers of weights (Fig. 3.4b). The first layer of neurons serves only as a distribution layer. Each neuron in subsequent layers produces NET and OUT signals. A neuron is associated with a set of weights that connects it with the input.

An Overview of Training

The objective of training is to adjust the weights so that application of a set of inputs produces the desired set of outputs. These input output set can be referred to as vectors. Each input vector is paired with a target vector representing the desired output and together these are called

a training pair. Usually a network is trained over a number of training pairs and the group of training pairs is called a training set.

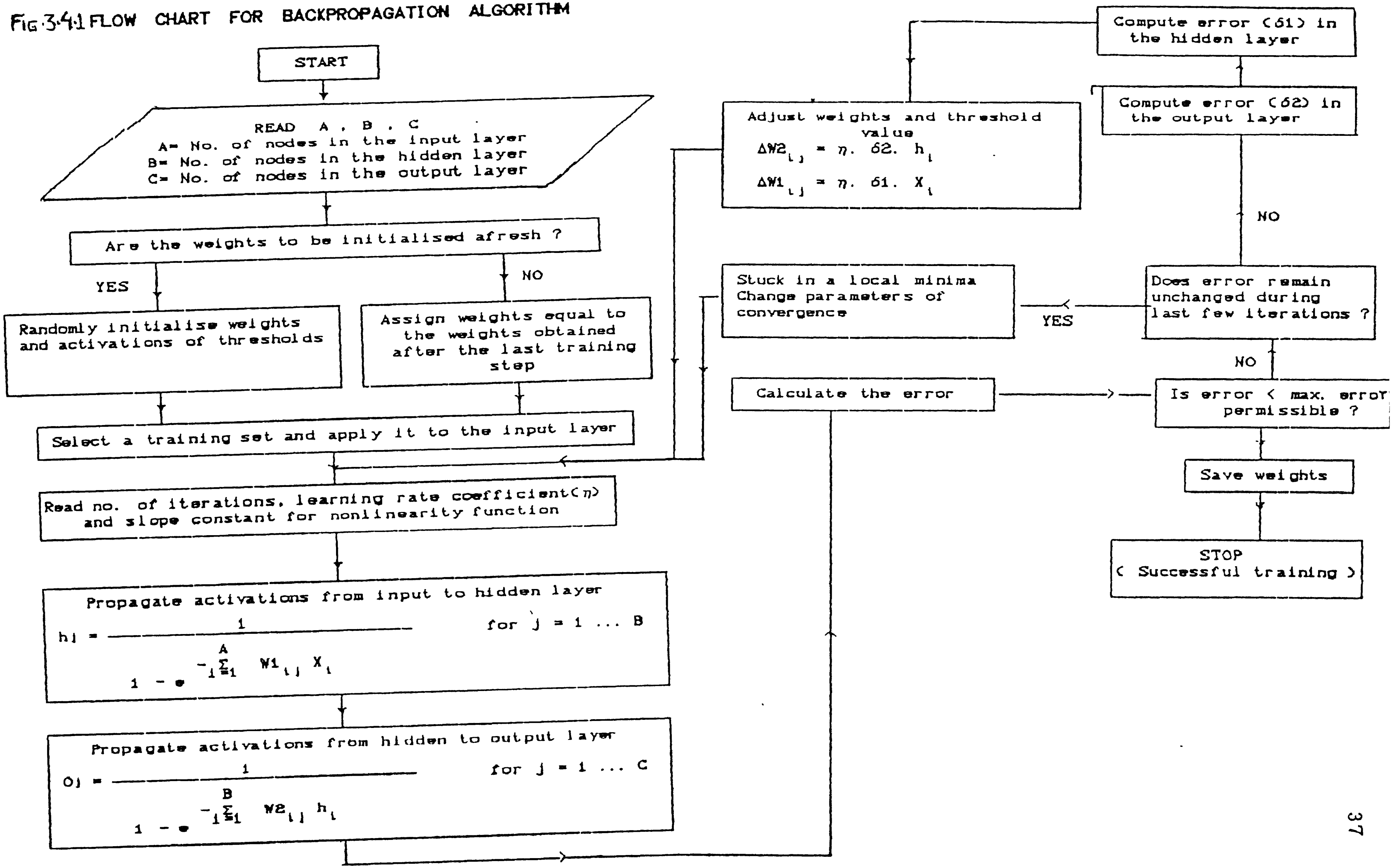
Before starting the training process all the weights must be initialized to small random numbers which ensures that the network is not saturated by large values of weights.

Following steps are followed for training the backpropagation network (A flow chart for backprpagation has been shown in Fig. 3.4.1) :

1. The next training pair is selected from the training set and the input vector is applied to the network.
2. The output of the network is calculated.
3. The error between the network output and the desired output (The target vector from the training pair ' is calculated.
4. The weights of the network are adjusted in a way that minimizes the error.
5. The steps from 1 through 4 are repeated for each vector in the training set until the error for the entire set is acceptably low.

The operation required in the steps 1 & 2 above are similar to the way in which the trained network will ultimately be used i.e., an input vector is applied and the resulting output is calculated.

Fig. 3.4.1 FLOW CHART FOR BACKPROPAGATION ALGORITHM



Forward pass

This consists of Steps 1 & 2 in which the signal propagates from input to output.

Reverse pass

This can be studied under two heads,

(a) Adjustment of the weights of the output layer (Fig. 3.4c)

Adjusting associated weights in this layer is easy because a target value is available for each neuron in this layer thereby making possible the use of Delta rule slightly modified.

If, for example, a single weight from a neuron p in the hidden layer j to a neuron q in the output layer k is to be trained then the output of a neuron in layer is subtracted from its target value to produce an error signal which is then multiplied by the derivative of the squashing function calculated for that neuron of the layer k, thereby producing the δ value,

$$\delta = OUT\ (OUT - 1)\ (Target - OUT) \tag{3.14}$$

the modification or change in weight is given by ,

$$\Delta W_{pq,k} = \eta\ \delta_{q,k}\ OUT_{p,j} \tag{3.15}$$

therefore,

$$W_{pq,k}\ (n+1) = W_{pq,k}\ (n) + \Delta W_{pq,k} \tag{3.16}$$

where,

$W_{pq,k}(n)$ = the value of weight from neuron p in the hidden layer to neuron q in the output layer at step n (before adjustment)

$W_{pq,k}(n+1)$ = value of the weight at step (n+1) after adjustment

$\delta_{q,k}$ = the value of δ for neuron q in the output layer k

$OUT_{p,j}$ = the value of OUT for neuron p in the hidden layer j

It should be noted that subscripts p and q refer to specific neurons while j and k refer to layers.

(b) Adjustments of weights of Hidden layers (Fig. 3.4d) :

Since the hidden layers have no target vectors, so the training process mentioned above can't be used. Backpropagation provides a workable algorithm for this which trains the hidden layers by propagating the output error back through the network layer by layer, adjusting weights at each layer. The Equations (3.15) and (3.16) are also applicable here, but for the hidden layers δ must be generated without a target vector. Thus,

$$\delta_{p,j} = OUT_{p,j}\ (1-OUT)\ (\textstyle\sum_q \delta_{q,k}\ W_{pq,k}) \qquad (3.17)$$

The above equation implies that, first, δ is calculated for each neuron of the output layer which is used to adjust weights feeding into output layer. It is then propagated back

through the same weights to generate value of δ for each neuron in the hidden layer which, in turn, are used to adjust the weights of the hidden layer.

Caveats

Despite many successful applications of the Backpropagation algorithm there are certain drawbacks :

* Long uncertain training process which may be due to non optimum step size.
* Outright training failures which may be from two sources viz. Network paralysis (training coming to a standstill due to large weights in region of small derivative of squashing function) and Local minimas.

3.4 NETWORK BIAS

All but the simpelest of recognition systems are unlikely to be perfect. Taking the visual system as an example, perfect recognition implies that an animal unerringly reacts to all images of the correct object (or class of objects) and never reacts to all other inappropriate images. But a recognition mechanism can only be expected to react appropriately to those images it has been selected to identify. One can not predict with certinity how an animal will react to new images it experiences. Many will have no effect, but because there is an almost infinite number of

possible images that the retina may experience, it is expected that some of these will elicit a greater response than the particular signals to which the system has been selected to respond. This has a clear analogy to the Darwin's Principle of Natural Selection. The question of why such preferences evolve remains a controversial issue. The mechanisms concerned with signal recognition posses inevitable biases in response that act as important agents of selection on signal form (Magnus Enquist and Anthony Arak, 1993).

CHAPTER 4

IMAGE PROCESSING

4.1 3D OBJECT RECOGNITION USING ANNs

The problem of recognition, mainly speech and shape recognition, is, nowadays, one of the most challenging areas of research. The main goal of computer vision research is to give computers human like visual capabilities so that the machine may :

* sense the environment in the field of view
* understand what is being sensed
* take appropriate actions as programmed

Desired characteristics of an ideal vision system

(a) It should be possible for the system to analyse scenes quickly and correctly.

(b) The system must be capable of handling sensor data from arbitrary viewing direction i.e., it requires a view independent modeling technique.

(c) The system must handle arbitrary complicated real world objects without giving preference to either curved or planar surfaces.

(d) It should be able to modify the world model data in order

to handle new objects and new situations.

(e) The system must be capable of handling a certain amount of noise in the sensor data without a significant degradation in the system performance.

The above characteristics can be acquired by an Artificial Neural Network combined with a good Vision System.

PROBLEM FORMULATION:

If an object is given which we have never seen before, we usually start to gather information about the object from different view points: gathering such information and storing the same is called as "model formation". Once familiar with many objects we can identify them from an arbitrary viewpoint without further investigations. Thus the central issue of vision is *Identification and Location of objects in the environment* which involves the critical step of linking of incoming visual information to stored object description.

A good vision system should be an autonomous single arbitrary view. The autonomous single arbitrary view 3D object recognition problem is to locate and identify 3D objects in the environment autonomously using a single and arbitrary view.

OBJECT RECOGNITION: This requires the determination of the translation parameters with respect to a known co-ordinate system as well as some orientation angles.

IDENTIFICATION: to get the shape of the object and to match it with the shapes which are stored in the database.

A given vision system if it can solve the stated 3D object recognition problem successfully can be extremely useful in a wide variety of applications including autonomous vehicle navigation and automatic inspection assembly.

RECOGNITION SYSTEM COMPONENTS (FIG. 4.1a)

How can one recognise something unless one knows what one is looking for. Hence perception is possible only if we have a model of the real world which therefore forms an essential component of object recognition system.

The World Model Module:

Perception is possible only if we have a model of the world, the popular models been classified into two: (a) high level or specialised models which have been developed mainly by work in the area of CAD viz., models of manmade components etc; and (b) low level models of image formation also called point wise models which have mainly been developed by work in the field of photometry where well worked out and easily adaptable models of the image formation process are provided.

The problems faced by both the types as in the case of a general purpose robot can be solved by using "parts and process model" which is an intermediate one between the two.

A good vision system should be capable of learning new types of objects. It should function even if the number of objects to be recognised is large, if the objects are occluded or largely obscured, or even when there are many unknown objects that are present in the scene. Moreover it should not employ specific object models.

Note: To build the world models, two theories in common use are that of superquadrics and fractals which are not being discussed here in detail for the sake of brevity.

The Sensor Data:

The input image in a computer vision system usually consists of numerical valued pixel arrays, where the pixel values are gray levels. In recent past, digitised range data have become available and its quality is improving day by day. Range data are often available in the form of array of numbers which are referred to as a range image (or depth map). Here the numbers quantify the distance from the sensor focal plane to object surfaces within the field of view along rays eminating from points on a regularly spaced grid.

The main advantage of using *Range imagery* (over the *Intensity imagery*) is simple separation of figure from the background which simplifies, relatively, the bottomup learning of object descriptions. However, finding the correct part structure remains as difficult as in the case of

intensity imagery.

The Symbolic Description:

The sensor data are processed until they reach the form of symbolic scene description. The model data can also be trnsformed into a symbolic scene description. A matching procedure can then be carried out on the quantities in this intermediate (symbolic scene description) domain which are referred to as features.

Interaction and mapping between different components of a recognition system have been shown in Figure. 4.1a.

Mapping (I): It creates intensity or range data.

Description process (D): acts on the sensor data and extracts relevant application independent features.

Modelling process (M): provides object models for real world objects.

Understanding or recognition process (U): involves an algorithm to perform matching between model and data descriptions.

Rendering process (R): produces synthetic sensor data from object models. Rendering provides an important link because it allows an autonomous system to check on its own understanding of the sensor data by comparing synthetic images to the sensed images.

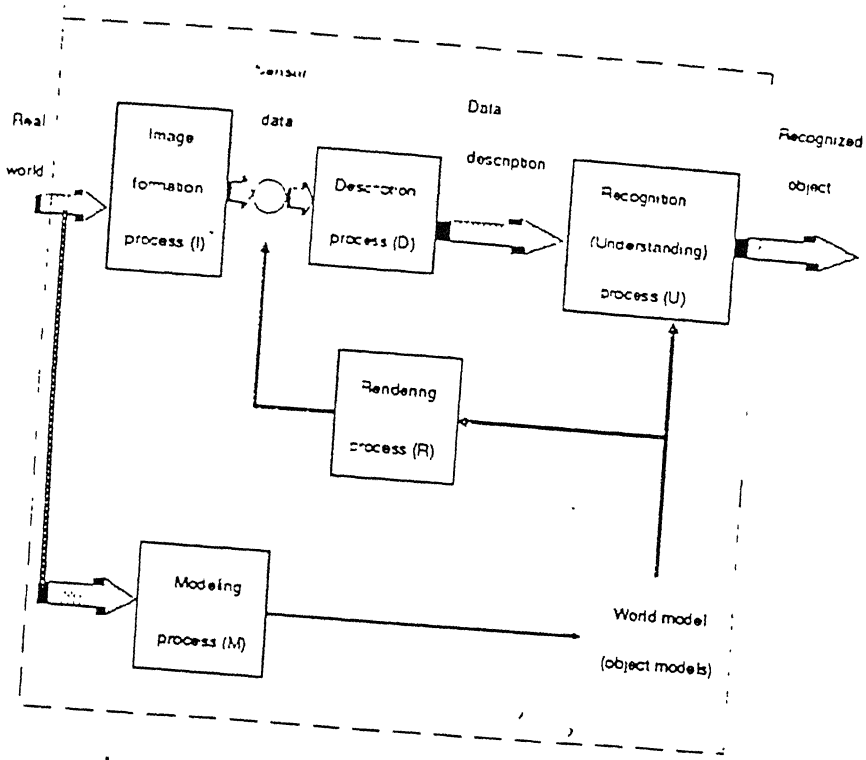


FIG. 4.1(a) BLOCKS OF A 3D RECOGNITION SYSTEM

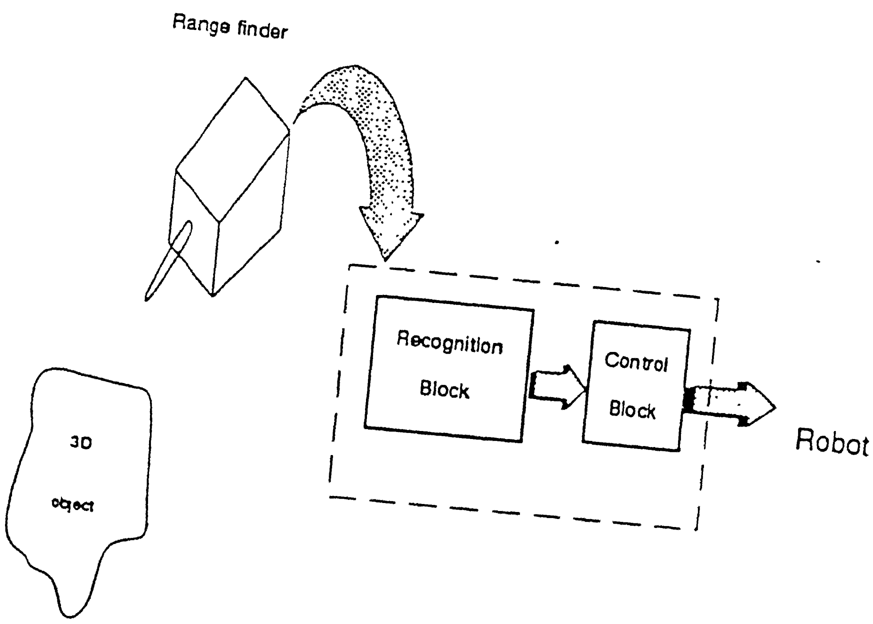


FIG. 4.1(b) THE PROPOSED OBJECT RECOGNITION SYSTEM

THE PROPOSED OBJECT RECOGNITION SYSTEM (FIG. 4.1b)

The essential components of this are:

(i) The laser range finder (the use of which has already been described)

(ii) The Recognition Block: Image pixel by themselves can determine nothing. It is necessary to have a model of image formation in order to obtain any assertion about the viewed scene. Hence the need of a model can't be sidestepped. The models used are based on the theory of fractals and superquadrics.

The ability of ANN to learn from experience, to generalize on their knowledge, and to perform abstraction make them suitable to be used in conjunction with a Vision system.

In fact, vision is the most remarkable of all of our intelligent sensing capabilities through which we are able to acquire information about our environment without direct contact. To the surprise of most of us, a TV Camera has a resolution on the order of 500 parts per sq. cm., while the human eye has a limiting resolution on the order of some $25x10^6$ parts per sq. cm., thus humans have a resolution 10,000 times fine than that of a TV Camera. It was, therefore, necessary to examine the processes and the problems involved in building computer vision systems which

share some similarities with human vision system and the ultimate objective is to determine a high level description of a 3D scene with a competancy level comparable to that of a human vision system.

It is very necessary to distinguish between a scene and an image of a scene. A scene is the set of physical objects in a picture area whereas an image is the projection of the scene on to a 2D Plane. Thus, a typical computer vision system should be able to perform the following operations:

1. Image formation, Sensing and Digitization.
2. Local processing and Image segmentation.
3. Shape formation and Interpretation.
4. Semantic analysis and description.

4.2 OVERVIEW OF VISION PROCESSING

The input to a vision system is 2D Image collected on some form of light sensitive surface. This surface is scanned by some means to produce a continuous voltage output that is proportional to the light intensity of the image on the surface. The output voltage f(x,y) is sampled at a discrete number of x and y points or pixel (picture element) positions and converted to numbers. The numbers correspond to the gray level intensity for black and white images. For colour images the intensity value is comprised of three separate arrays of numbers, one for the intensity value of each of the basic

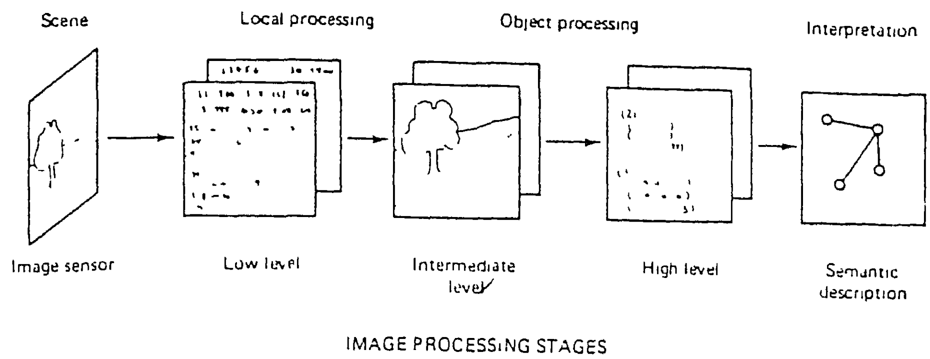


Figure 4.2(a) Processing stages in computer vision systems

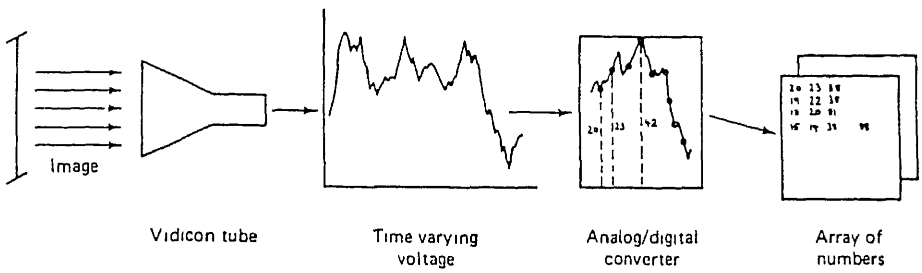


Figure 4.2(b) Transforming the image to numbers

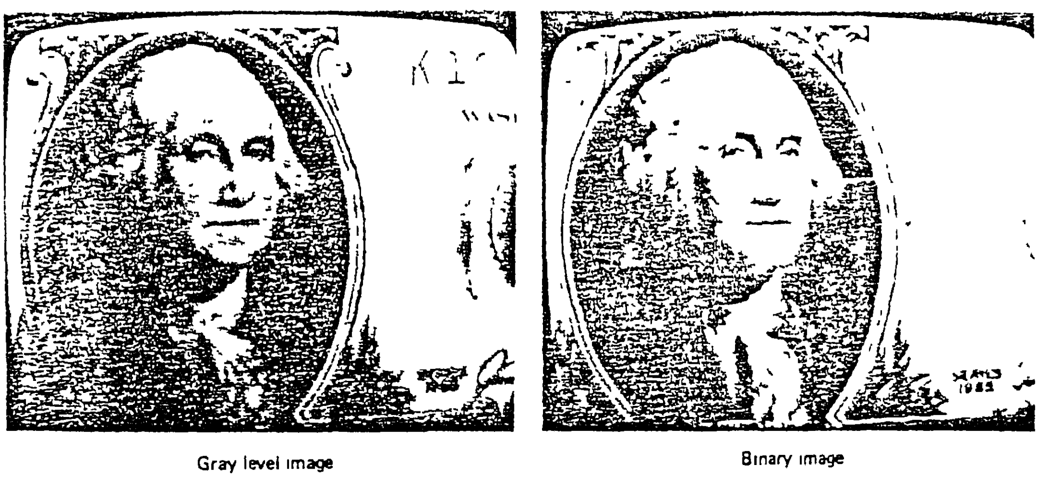


Figure 4.2(c) Threshold transformation of an image

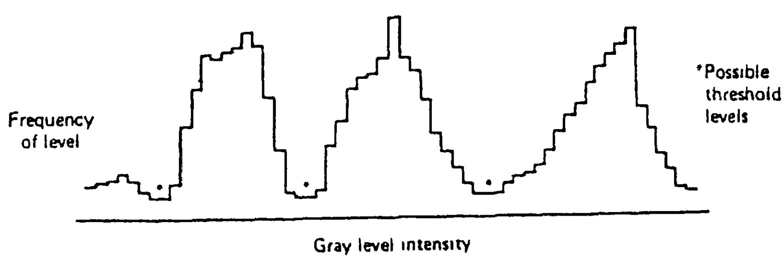


Figure 4.2(d) Histogram of light intensity levels.

colours (Red, Green, and Blue).

Thus, through the digitization process the image is transformed from a continuous light source into an array of numbers which correspond to the local image intensities at the corresponding x-y pixel positions on the light sensitive surfaces.

Using the array of numbers certain low level operations are performed such as a smoothing of neighbouring points to reduce noise, finding outlines of objects or edge elements, thresholding (recording maximum and minimum values only depending on some fixed intensity threshold level) and determining texture, colour and other object features. These initial processing steps are the ones which are used to locate and accentuate object boundaries and other structures within the image.

Then next stage of processing, the intermediate level, involves connecting, filling in and combining boundaries, determining regions and assigning descriptive labels to objects that have been accentuated in the first stage. This stage builds higher level structures from the lower level elements of the first stage. After completion, it passes on labelled surfaces such as geometrical objects that may be capable of identification.

High level image processing consists of identifying the

important objects in the image and their relationships for subsequent descriptions as well as defined knowledge structures and hence for use by a reasoning component.

Special types of vision systems may also require 3D processing and analysis as well as motion detection and analysis.

4.3 OBJECTIVES OF COMPUTER VISION SYSTEMS

The ultimate goal of computer image understanding is to build systems that equal or exceed the capabilities of human vision systems. In an ideal case computer vision systems would be capable of interpreting and describing any complex scene in complete detail. But the amount of processing and the storage required to interpret and describe a complex scene can be enormous. For example, a single image for a high resolution aerial photograph may result in some four to nine million pixels (Bytes) of information and require on the average some ten to twenty computations per pixel.

4.4 IMAGE TRANSFORMATION AND LOW LEVEL PROCESSING

This includes the process of forming an image and transforming it to an array of numbers which can then be operated on by a computer. In this first stage only local processing is performed on the numbers to reduce noise and other unwanted picture elements and to accentuate object boundaries.

TRANSFORMING LIGHT ENERGY TO NUMBERS :

First stage in image processing requires a transformation of light energy to numbers, the language of computers. To accomplish this some form of light sensitive devices (Transducers) are used such as a Vidicon Tube or Charge Coupled Device (CCD).

A Vidicon Tube is the type of sensor typically found in home or industrial Video systems. A lens is used to project the image on to a flat surface of the vidicon. The Tube surface is coated with a photoconductive material whose resistance is inversely proportional to the light intensity falling on it. An Electron Gun is used to produce a flying spot scanner with which to rapidly scan the surface left to right and top to bottom. The scan results in a time varying voltage which is proportional to the scan spot image intensity. The continuously varying output voltage is then fed to an analog to digital converter (ADC) where the voltage amplitude is periodically sampled and converted to numbers. A typical ADC unit will produce thirty complete digitized frames consisting of 256x256 or 512x512 (or more) samples of an image per second. Each sample is a number (or triple of numbers in the case of colour systems) ranging from 0 to 64 (6 bits) or 0 to 255 (8 bits). The image conversion process has been shown in the Figure 4.2b.

A CCD is typical of the class of solid state sensor devices known as charge transfer devices that are now being used in many vision systems. A CCD is a rectangular chip consisting of an array of capacitative photo detectors, each capable of storing an electrostatic charge. The charges are scanned like a clock driven shift register and converted into a time varying voltage which is proportional to the incident light intensity on the detectors. This voltage is sampled and converted to integers using an ADC unit as in the case of the vidicon tube. The density of the detectors on the chip is quite high. For example, a CCD chip of about 5 sq. cm. in area may contain as many as 1000x1000 detectors.

The numeric outputs from the ADC unit are collected as array of numbers which corresponds to the light intensity of the image on the surface of the transducer. This is the input to the next stage of processing.

PROCESSING THE QUANTIZED ARRAYS:

The array of numbers produced from the image sensing device may be thought of as the lowest, most primitive level of abstractions in the vision understanding process. The next step in the processing hierarchy is to find some structures among the pixels such as pixel clusters which define object boundaries or regions within the image. Thus it is necessary to transform the array of raw pixel data into regions of

discontinuities and homogeneity, to find edges and other delimiters of of these object regions. A raw digitized image will contain some noise and distortion, hence computations to reduce these may be necessary before locating edges and regions. Other low level operations include thresholding to help define homogeneous regions, and different forms of edge detection to define boundaries.

THRESHOLDING:

This is the process of transforming the gray level representation to a binary representation of the image. All digitized array values above some threshold value T are set equal to the maximum gray level value (black) and value less then or equal to T are set equal to zero (white). Thus thresholding is one way to segment the image into sharpened object regions by enhancing some portions and reducing others like noise and other unwanted features. Threshclding at different levels may be necessary to handle extreme intensities. The best choice of threshold value can be decided by a study of histogram of light intensity levels (Fig.4.2c).

SMOOTHING: (Fig. 4.3a)

It is a form of digital filtering. It is used to reduce noise and other unwanted features and to enhance certain image features. Techniques involved are many, some of which

are local averaging, use of models and parametric form fitting.

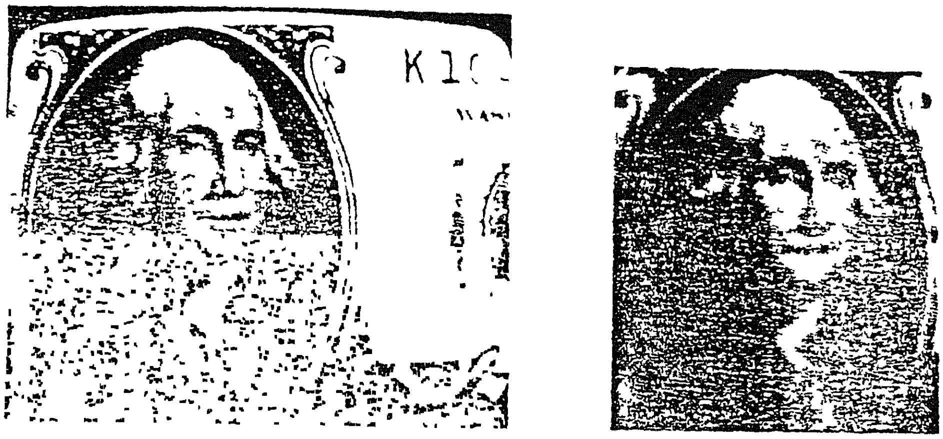
LOCAL EDGE DETECTION:

It is a process of finding a boundary or delimiter between two regions. An edge will show up as a relatively thin line or arc which appears as a measurable difference in contrast between otherwise homogeneous regions. Several approaches have been proposed for locating the boundaries:

* Boundaries separating adjoining regions represent a discontinuity in one or more of features, such as colour, texture, three dimensional flow effects or intensity, a fact which can be exploited by measuring the rate of change of a feature value over the image surface.

* There is some evidence to support the belief that the *human eye* uses a form of Gaussian transformation called lateral inhibition which enhances the contrast between gradually changing objects like an object and its background.

* Another approach used to filter the digitized image applies frequency domain transforms such as the Fourier Transforms. Since edges represents higher frequency components the transformed image can be analysed on the basis of its frequency. An efficient algorithm called as *Fast Fourier Transform* has been developed which when applied to an array of intensity values produces an array of complex



Original image array

Smoothed image array

Figure 4.3(a) Application of a smoothing mask

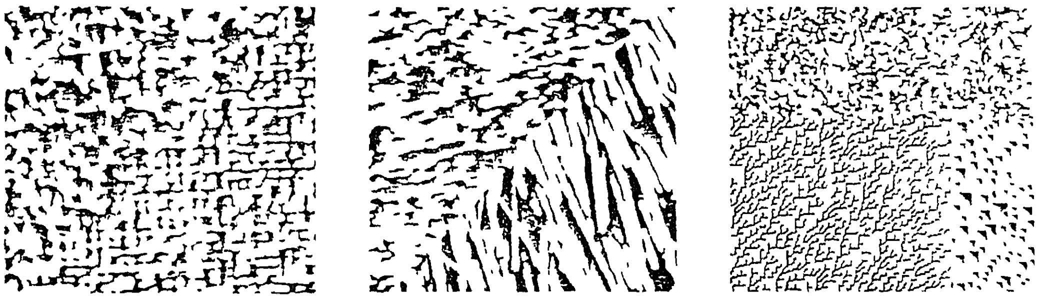


Figure 4.3(b) Examples of textured surfaces

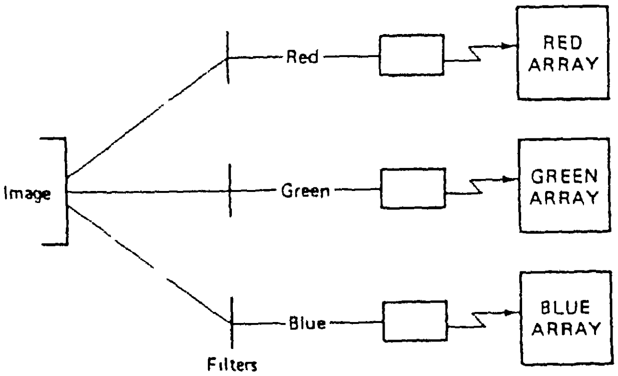


Figure 4.3(c) Color separation and processing

numbers that correspond to spatial frequency components of the image.

* Another method is that of model fitting: a model in the form of a mask is shifted over a region and compared to the corresponding gray levels.

Texture and color are also used to identify boundaries: *Texture* is a repeated pattern of elementary shapes occurring on an object's surface. The structure in texture is generally too fine to be resolved, yet still coarrse enough to cause noticeable variations in the gray levels (Fig 4.3b). These methods of analysis which have been developed are based on. (a) the application of pattern matching; (b) the use of Fourier Transform; and (c) modeling with special functions known as fractals.

Color when used to identify regions requires more than three times as much processing as gray level processing because the image must, first of all, be separated into three primary colors (Fig.4.3c). In complex scene analysis color may be the most effective method of segmentation and object identification.

4.5 INTERMEDIATE LEVEL IMAGE PROCESSING

This concentrates on segmenting the image surface into larger global structures using homogeneous features in pixel regions and boundaries formed from pieces of edges discovered

during the low level processing.

This level requires that pieces of edges be combined into contiguous contours which form the outline of objects, partitioning the image into coherent regions, developing models of segmented objects and then assigning labels which characterize the object regions. The general process of forming contours is called *segmentation*, the methods employed for which are: (i) Graphical edge finding; (ii) Edge finding with Dynamic programming; and (iii) Region segmentation through splitting and merging.

Once the image has been segmented into disjointed regions, their shapes, spatial interrelationships, and other characteristics can be described and labeled for subsequent interpretation which may be 2D or 3D 'scene description'.

Template Matching is the process of comparing patterns found in an image with prestored templates that are already named. The matching process may occur at lower levels using individual or group of pixels using correlation techniques or at higher image processing levels using labeled region structures.

Template matching can be effective only when the search process is constrained in some way, for example, the types of scenes and permissible objects should be known in advance, thereby limiting the possible pattern-template pairs.

4.6 HIGH LEVEL PROCESSING

High level processing techniques are less mechanical then either of the former two. In this the intermediate level region descriptions are transformed into high level scene descriptions in one of the knowledge representation formalism like associative nets, frames, FOPL statements etc.

The end objectives of this stage is to create high level knowledge structures which can be used by an inference programme. It is obvious that the resulting structure should uniquely and accurately describe the important objects in an image including their interrelationship. In fact this particular vision application will dictate the appropriate level of detail and what is considered to be important in a scene description.

Before a scene can be described in terms of high level structures, prestored model descriptions of the objects must be available which are compared with the region description created during the intermediate level stage.

Associative networks have become a popular method of scene descriptions since they show the relationship among the objects as well as object characteristics.

CHAPTER 5

PRESENT STUDY

5.1 DATA AVAILABLE

The form in which the data is and can be available to us does effect the planning for future developmental activity. A very common method in current practice for keeping a record of the field traffic data i.e., of the incidents occuring on a road stretch, is the preparation of video cassettes by operating a video camera sitting in a test vehicle so as to have an overall view of traffic interaction. A large number of such cassettes have been prepared by CRRI, New Delhi on various National and State Highways (during January 1992 to April 1993). A special Video Instrumentation System developed by Australian Road Research Board is mounted on a test vehicle which is a Maruti van in this case.

The essential components of the Instrumentation system, without going into much details, are:

(i) Video cameras- one at the front to view the oncoming vehicles and one at the rear for viewing at the back.

(ii) Video recorder

(iii) Video monitor

(iv) Video splitter (multiviewer) which enables the pictures taken by more than one video camera to be recorded in a single picture frame i.e., for viewing the pictures taken by the two cameras simultaneously it divides the monitor frame into two parts horizontally (or vertically). It can even divide the monitor screen into four parts.

(v)Video timer

These may be supplemented with:

(vi)Rotopulser which is used for vehicle interaction studies. In this, pulses are sent, counted, and sent to the video clock for measuring distance along the road when the instrumented vehicle is run. This replaces the inaccurate odometer and gives the distance co-ordinates in metres (Least count is 1 m.).

(vii)Speedo odometer

(viii)Radar speedmeter which is used to detect the speed of a moving vehicle.

The first line of the digitized data on screen indicates the date in year, month, and day followed by time in hours and minutes. The last figure of the second line completes the time indication in tenth of a second. The other figures in the second line indicate relative speed of the

oncoming vehicle with reference to the test vehicle (in Kmph) , speed of the test vehicle (in Kmph) and distance coordinate of the test vehicle (in m). Thus it is in this form the field data is and can be available to us.

5.2 MANUAL IMAGE PROCESSING

In order to obtain various parameters and to establish relationship between different parameters related to traffic interaction, presently the video cassettes are being processed manually. The instrument setup for this has been shown in the Fig. 5.1(a).

The signal from the VCR goes to the Video scalar from where it goes to the video splitter which passes it on to the video monitor. The video scaler helps creating a scale/grid on the screen to facilitate the noting down of the width of the vehicle and the splitter helps divide the screen into two parts to watch the views from the two cameras simultaneously. To note down the reading as soon as a clear picture of an oncoming vehicle comes in the field of view of the front camera the scene is paused and with the help of video scaler the image size is noted. There are polynomials calibrated for directly getting the distance of the oncoming vehicle from the test vehicle knowing the image size (Fig. 5.1b). If at

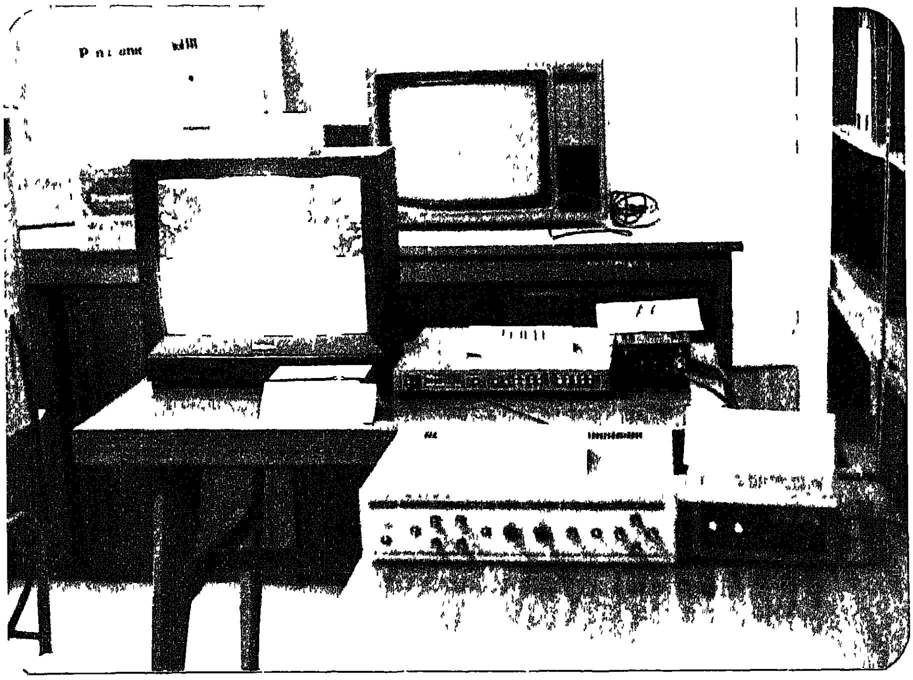


Fig. 5.1(a)Instrument set up for manual data processing

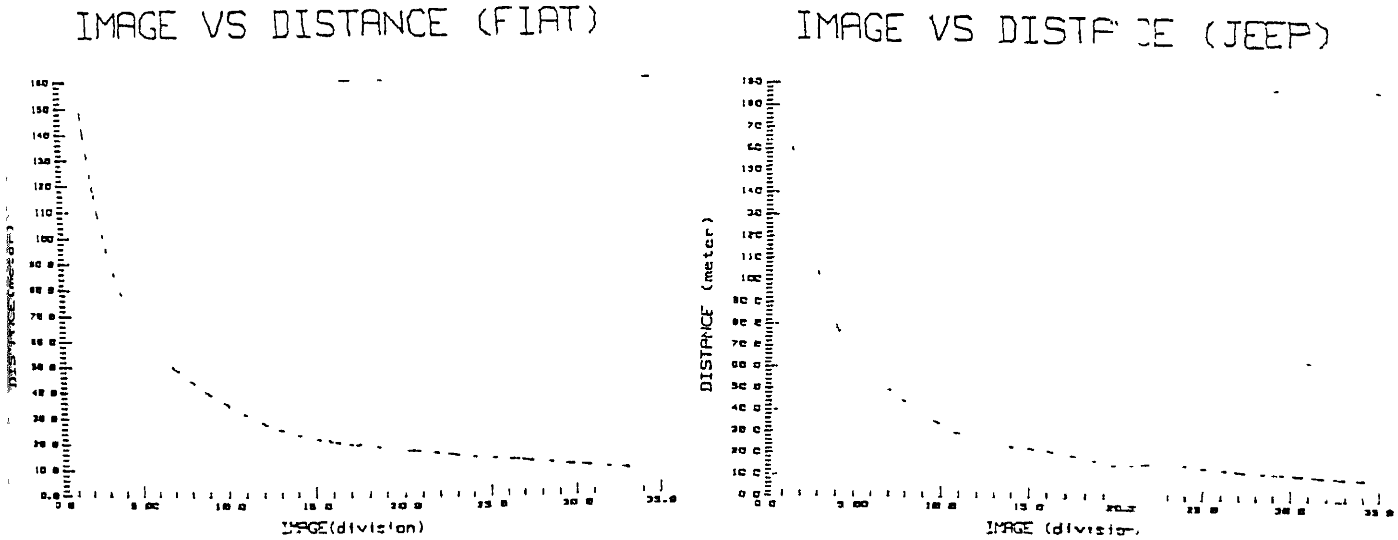


Fig. 5.1(b) Image size vs Distance Polynomials

the first instance the image size gives the distance d1 at the time t1 and then another image size gives the distance as d2 at time t2, we can have the relative speed of the vehicle, = (d1 - d2)/(t1 - t2)

The calculation of the parameters such as velocity etc. help improve the safety. Moreover a study of these cassettes help in classifying the various vehicle types running through particular National Highways, State Highways and other roads for prediction of future traffic.

5.3 AUTOMATIZED IMAGE PROCESSING USING ANNs

For improved design of transportation facilities if such a huge amount of data is processed manually enormous amount of time shall be required. Hence, our basic purpose is to automatize this process of image processing so as to help in the design of automatized intersection control, automatized accident warning systems etc.

To begin with this challenging task the first step that has been planned is to use the ANNs in the recognition of vehicles from their 3D scene (after converting the 3 D scene into a 2D digitized image). The conversion of 3 D scene to a 2D digitized one has to be through a computer vision system. Even if, only recognition of the various vehicles that form

the traffic stream is successfully possible much of the manual work gets reduced and achievement of higher goals through this automatization (involving dynamics also) seems possible.

It will not be out of place to mention that the digitization of the image of vehicles can also be done in the form of a 8x8 matrix or 10x10 matrix by using video scaler which forms grids on the screen enabling us to adjust the vehicle image (front views at different angles have been chosen in our study) within 8x8 or 10x10 square blocks formed on the screen (Fig. 5.2a). and then assigning value 0 to blocks not containing any part of the vehicle and 1 to blocks which contain any part. The instrument set up for this type of digitization as been shown in Fig. 5.2b.

5.4 SOFTWARE-ITS INPUT& OUTPUT

A software has been developed using the Backpropagation algorithm for ANN training (Fig. 5.3 shows the model) in which dimensions have been so specified that it can work for 100 input, 4500 hidden layer and 3 output nodes; if the no. of inputs is to be increased the dimensions can be changed accordingly but it has been found by trial and error that even this much number of nodes require a very large memory because as a convention 'n' input nodes require 'n(n-1)'

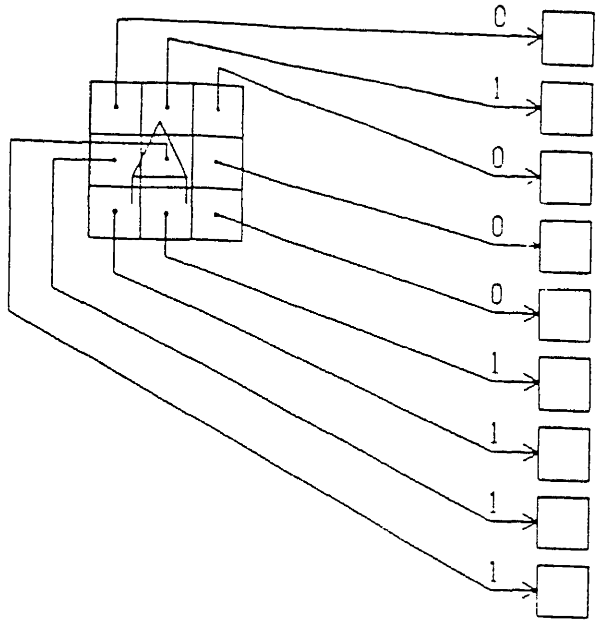
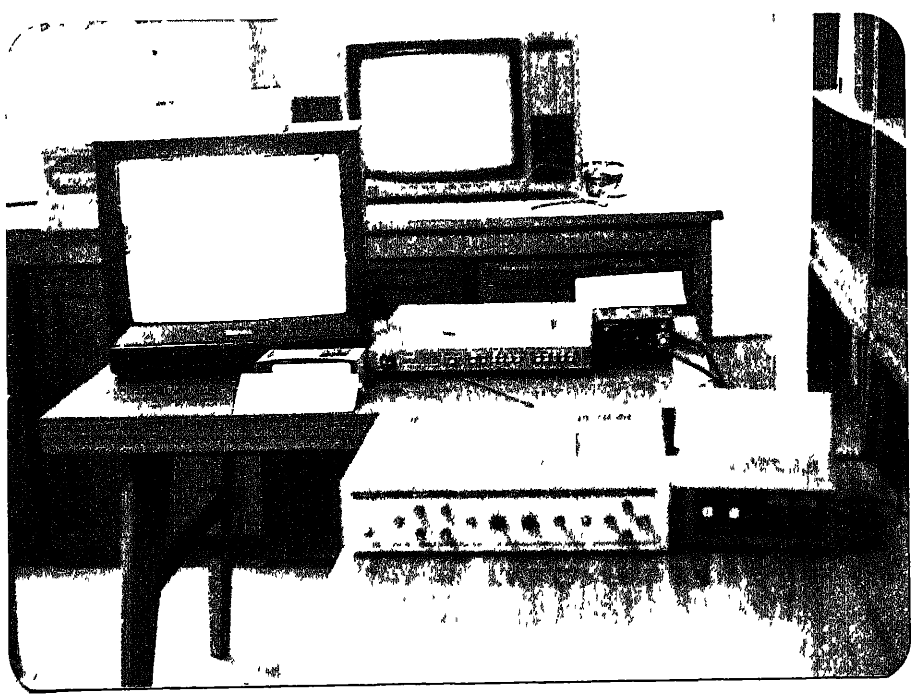


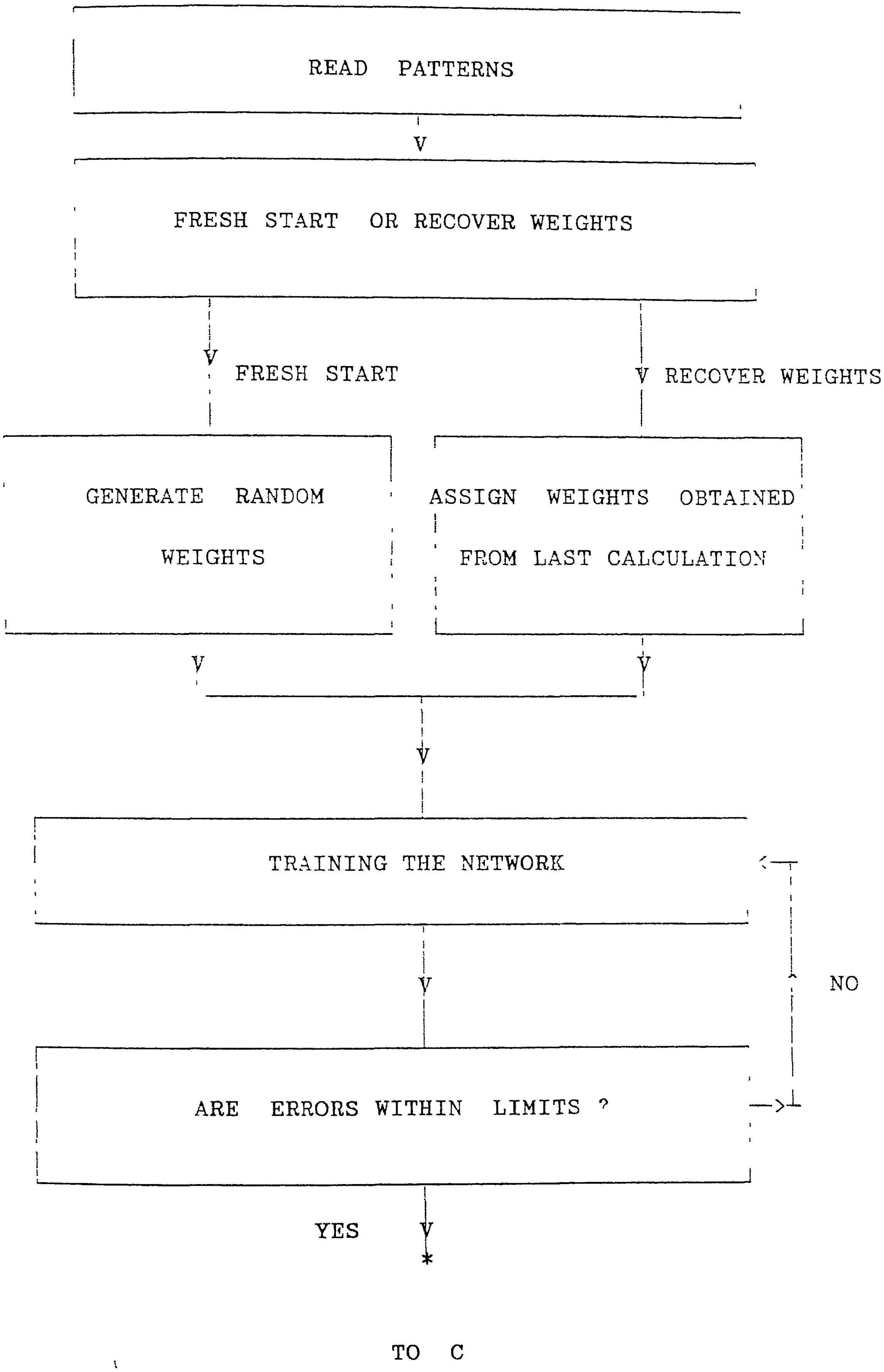
Fig. 5.2(a) Image digitisation to obtain patterns.



hidden layer nodes, hence the large no. of weights can be imagined. Hence, higher is the number of input nodes the larger is the memory that is required to be allocated on the machine. It is because of this that a small 8*8 matrix (which means 64 input elements) has been preferred to train the network with. It has been found that a 512x512 matrix will require about 100 MB of memory for training the network and imposes a serious limitation. For any particular vehicle the digitized front views at different angles form the input vectors and all of these have a common output vector which may be 0, 1, 2, 3 etc. expressed in the binary form. Thus, as many training pairs for each vehicle are available as are the number of digitized front views at different angles. The aforesaid practical difficulties could only be known after an extensive trial and error process during the training of the network.

The working of the different modules as shown in Fig. 5.3 is as follows:

Readpatterns: This part of the program reads the training set consisting of several input-output training pairs and stores it so that there is no need of going through the patterns again and again. The number of nodes in the input, output, and hidden layer are specified in this phase and the patterns are read accordingly.


READ PATTERNS
FRESH START OR RECOVER WEIGHTS
FRESH START
RECOVER WEIGHTS
GENERATE RANDOM WEIGHTS
ASSIGN WEIGHTS OBTAINED FROM LAST CALCULATION
TRAINING THE NETWORK
ARE ERRORS WITHIN LIMITS ?
NO
YES
TO C

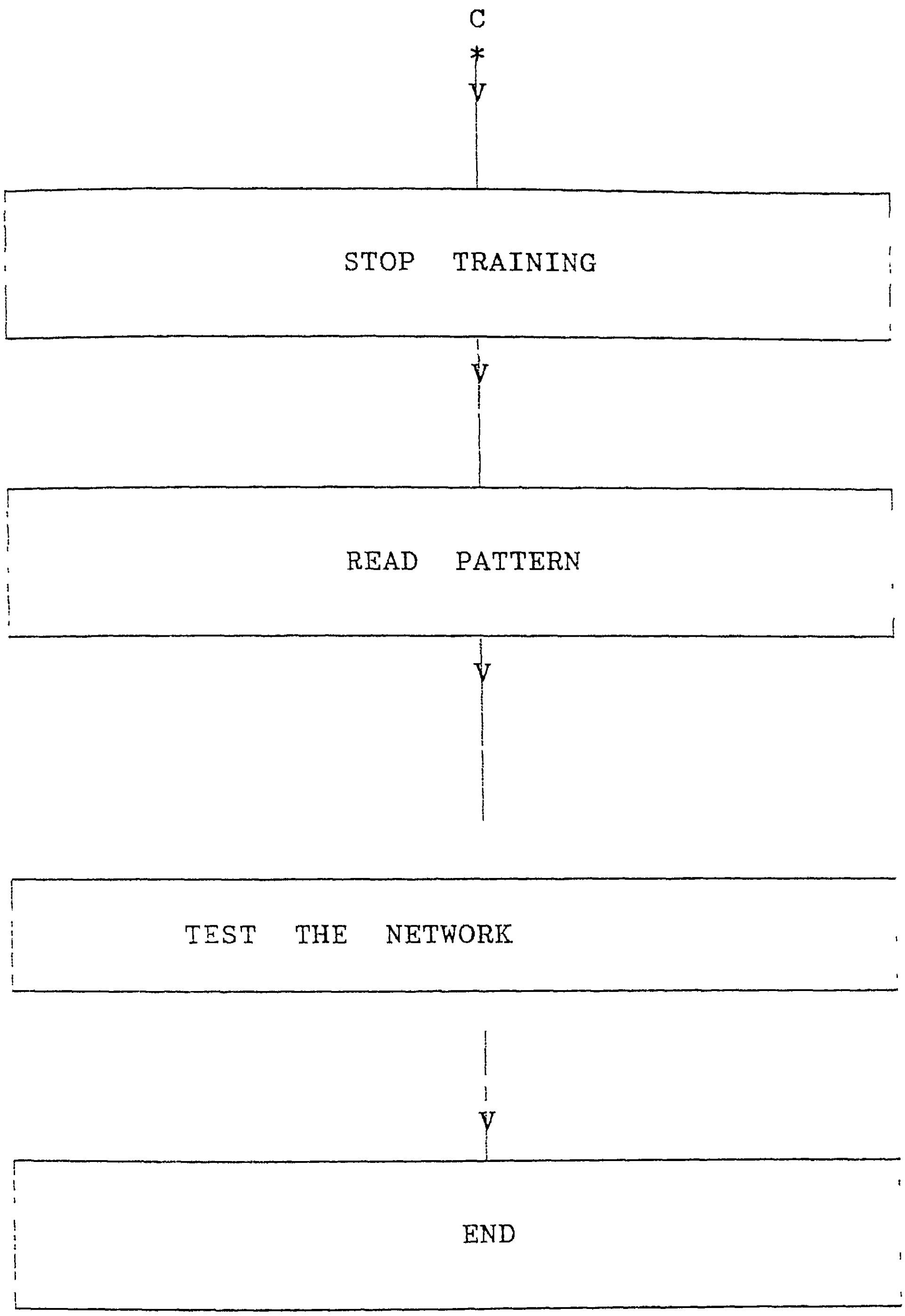


FIG. 5.3 MODEL OF NEURAL NETWORK SOFTWARE

Fresh start/Recover weights: Here it has to be fed to the software whether we want to initialise the weights i.e., start afresh or we want to recover the weights of the previous calculation. If we choose the first option the weights are initialised by random number generation otherwise weights of previous calculation are recovered.

Training: This phase of the program uses Backpropagation algorithm for modifying the weights again and again, depending on the error function, till the percentage error is within permissible limits i.e., the network reaches a state in which it can give correct output when exposed to an input pattern similar to the ones it has been trained with.

Testing: In this phase a similar inpt is fed and the output obtained. It is then seen whether this output is right. If it is so, the proper working of the network is proven. But, this phase requires an extra pattern.

The neural network was trained with the digitized inputs and the training procedure was quite cumbersome because even 64 (8x8 matrix) input and 1000 hidden nodes mean a large number of weights which are to be modified reiteratively for the prescribed number of times (say 500 or 1000) and this process is repeated again and again after changing the parameters of convergence each time till the error reaches an acceptably low value. The results of training by 4 training sets follow.

Such results favour the use of ANNs for pattern recognition of traffic elements.

	STEP NO		0		
given o/p	390000	cal o/p	392694	error	-3 340440 %
given o/p	540000	cal o/p	670617	error	-24 188305 %
given o/p	620000	cal o/p	760523	error	-22 665031 %
given o/p	510000	cal o/p	603435	error	-20 687019 %
given o/p	340000	cal o/p	335508	error	1 321198 %
given o/p	600000	cal o/p	746100	error	-24 349926 %
	STEP NO		1		
given o/p	390000	cal o/p	367510	error	3 286929 %
given o/p	540000	cal o/p	542571	error	- 476197 %
given o/p	620000	cal o/p	611142	error	1 428733 %
given o/p	510000	cal o/p	501187	error	- 237489 %
given o/p	340000	cal o/p	325795	error	3 886854 %
given o/p	600000	cal o/p	594612	error	898004 %
	STEP NO		2		
given o/p	390000	cal o/p	369300	error	2 815865 %
given o/p	540000	cal o/p	539934	error	012241 %
given o/p	620000	cal o/p	607924	error	1 947793 %
given o/p	510000	cal o/p	499483	error	103307 %
given o/p	340000	cal o/p	331737	error	2 430204 %
given o/p	600000	cal o/p	590304	error	1 616081 %
	STEP NO		3		
given o/p	390000	cal o/p	378906	error	287836 %
given o/p	540000	cal o/p	550765	error	-1 993510 %
given o/p	620000	cal o/p	619461	error	086965 %
given o/p	510000	cal o/p	510047	error	-2 009332 %
given o/p	340000	cal o/p	342822	error	- 829942 %
given o/p	600000	cal o/p	600799	error	- 131567 %
	STEP NO		4		
given o/p	390000	cal o/p	373321	error	1 757749 %
given o/p	540000	cal o/p	546163	error	-1 141318 %
given o/p	620000	cal o/p	616399	error	580770 %
given o/p	510000	cal o/p	504880	error	- 976062 %
given o/p	340000	cal o/p	338869	error	332769 %
given o/p	600000	cal o/p	596641	error	559817 %
	STEP NO		5		
given o/p	390000	cal o/p	375084	error	1 293805 %
given o/p	540000	cal o/p	547393	error	-1 369008 %
given o/p	620000	cal o/p	618013	error	320452 %
given o/p	510000	cal o/p	506147	error	-1 229310 %
given o/p	340000	cal o/p	342024	error	- 595214 %
given o/p	600000	cal o/p	597516	error	413954 %

O/P = OUTPUT

RESULTS OF FIRST TRAINING SET (TRIAL SET)

4 INPUT NODES, 12 HIDDEN NODES, 1 OUTPUT NODE

6 PATTERNS, 500 ITERATIONS

999509	-3 073321
-2 730206	902907
999508	-3 073321
-2 730206	902907
999509	- 898741
-2 730206	755332
999509	- 898741
-2 730206	755332
999503	- 898741
-2 730206	755332
999509	- 898741
-2 730206	755332
999509	- 898741
993225	755832
-2 392132	- 898741
930225	755932
-2 292132	
931225	- 898741
-2 393132	904020
956325	-2.810594
-2 392132	804020
721225	-2 310594
-2 392132	804020
936325	-2 310594
-2 392132	804020
936325	-2 310594
439741	904020
-2 495015	-2 910574
433741	804020
-2 495015	-2 310594
428711	8040.0
-2 495015	
433741	
-2 495015	
433741	
-2 4950,5	
433741	
-2 495015	
423711	
[illegible]	
-3 073821	
903907	
-3 073321	
902907	
-3 073321	
902907	
-3 073321	
902907	

CORRESPONDING WEIGHTS

STEP NO		0			
given o/p	277773	cal o/p	.022087	error	92 048836 %
given o/p	415152	cal o/p	.032321	error	92 214546 %
given o/p	576763	cal o/p	055682	error	90 345894 %
given o/p	295859	cal o/p	.021730	error	92 398361 %
given o/p	544445	cal o/p	057831	error	89 378059 %
given o/p	471717	cal o/p	042410	error	91 009430 %
given o/p	318182	cal o/p	024825	error	92 197815 %
given o/p	431313	cal o/p	036009	error	91 651291 %
STEP NO		1			
given o/p	277773	cal o/p	274329	error	1 241616 %
given o/p	415152	cal o/p	385744	error	7 083501 %
given o/p	576763	cal o/p	552468	error	4 213131 %
given o/p	285859	cal o/p	272160	error	4 791988 %
given o/p	544445	cal o/p	547621	error	- 583452 %
given o/p	471717	cal o/p	464029	error	1 630008 %
given o/p	318182	cal o/p	304318	error	4 357757 %
given o/p	431313	cal o/p	415272	error	3 719034 %
STEP NO		2			
given o/p	277773	cal o/p	287846	error	-3 624587 %
given o/p	415152	cal o/p	407182	error	1 919631 %
given o/p	576763	cal o/p	576318	error	078013 %
given o/p	285859	cal o/p	288016	error	-.754622 %
given o/p	544445	cal o/p	562047	error	-3 233036 %
given o/p	471717	cal o/p	480309	error	-1 821167 %
given o/p	318182	cal o/p	322864	error	-1.471637 %
given o/p	431313	cal o/p	438618	error	-1 693719 %
STEP NO		3			
given o/p	[illegible]	cal o/p	274495	error	1 181824 %
given o/p	[illegible]	cal o/p	394556	error	4 960893 %
given o/p	[illegible]	cal o/p	564655	error	2 100138 %
given o/p	[illegible]	cal o/p	275789	error	3 522657 %
given o/p	[illegible]	cal o/p	543185	error	231360 %
given o/p	[illegible]	cal o/p	463691	error	1 701564 %
given o/p	[illegible]	cal o/p	310625	error	2 374946 %
given o/p	[illegible]	cal o/p	425305	error	1 392734 %
STEP NO		4			
given o/p	[illegible]	cal o/p	280704	error	-1 053336 %
given o/p	[illegible]	cal o/p	404222	error	2 632666 %
given o/p	[illegible]	cal o/p	575493	error	221091 %
given o/p	[illegible]	cal o/p	282570	error	1 115346 %
given o/p	[illegible]	cal o/p	549673	error	- 960340 %
given o/p	[illegible]	cal o/p	471455	error	055641 %
given o/p	[illegible]	cal o/p	318477	error	- 092756 %
given o/p	[illegible]	cal o/p	434514	error	- 742113 %
STEP NO		5			
given o/p	277773	cal o/p	277559	error	.078707 %
given o/p	415152	cal o/p	401743	error	3 229809 %
given o/p	576763	cal o/p	573445	error	576011 %
given o/p	285859	cal o/p	.279799	error	2 119941 %
given o/p	544445	cal o/p	544222	error	.040857 %
given o/p	.471717	cal o/p	.467224	error	.952439 %
given o/p	318182	cal o/p	315586	error	.815742 %
given o/p	431313	cal o/p	430733	error	.134483 %

o/p = output

RESULTS OF SECOND TRAINING SET

10 INPUT NODES, 90 HIDDEN NODES, 1 OUTPUT NODE

8 PATTERNS, 1000 ITERATIONS

	STEP NO		0		
given o/p	0.100000	cal o/p	0 086453	error	13 547249 %
given o/p	0 100000	cal o/p	0 086453	error	13 547249 %
given o/p	0 100000	cal o/p	0 086453	error	13 547249 %
given o/p	0 100000	cal o/p	0 088361	error	11 639081 %
given o/p	0 100000	cal o/p	0 088361	error	11 639081 %
given o/p	0 100000	cal o/p	0 088361	error	11 639081 %
given o/p	0 100000	cal o/p	0 086663	error	13.336725 %
given o/p	0 100000	cal o/p	0 086663	error	13 336725 %
given o/p	0 100000	cal o/p	0 086663	error	13 336725 %
STEp	1				
given o/p	0 100000	cal o/p	0 100228	error	-0 228412 %
given o/p	0 100000	cal o/p	0.100228	error	-0 228412 %
given o/p	0 100000	cal o/p	0 100228	error	-0 228412 %
given o/p	0 100000	cal o/p	0 102472	error	-2 472214 %
given o/p	0 100000	cal o/p	0 102472	error	-2.472214 %
given o/p	0 100000	cal o/p	0 102472	error	-2 472214 %
given o/p	0 100000	cal o/p	0 100395	error	-0 395075 %
given o/p	0 100000	cal o/p	0 100395	error	-0 395075 %
given o/p	0 100000	cal o/p	0 100395	error	-0.395075 %
STEp	2				
given o/p	0 100000	cal o/p	0 099781	error	0 219308 %
given o/p	0 100000	cal o/p	0 099781	error	0 219308 %
given o/p	0 100000	cal o/p	0 099781	error	0 219308 %
given o/p	0 100000	cal o/p	0 101999	error	-1 998790 %
given o/p	0 100000	cal o/p	0 101999	error	-1 998790 %
given o/p	0 100000	cal o/p	0 101999	error	-1 998790 %
given o/p	0 100000	cal o/p	0 099918	error	0 081681 %
given o/p	0 100000	cal o/p	0 099918	error	0 081681 %
given o/p	0 100000	cal o/p	0 099918	error	0 081681 %
STEp	3				
given o/p	0 100000	cal o/p	0 100226	error	-0 226296 %
given o/p	0 100000	cal o/p	0 100226	error	-0 226296 %
given o/p	0 100000	cal o/p	0 100226	error	-0 226296 %
given o/p	0 100000	cal o/p	0 102430	error	-2 430417 %
given o/p	0 100000	cal o/p	0 102430	error	-2 430417 %
given o/p	0 100000	cal o/p	0 102430	error	-2 430417 %
given o/p	0 100000	cal o/p	0 100315	error	-0 314660 %
given o/p	0 100000	cal o/p	0 100315	error	-0 314660 %
given o/p	0 100000	cal o/p	0 100315	error	-0 314660 %
STEp	4				
given o/p	0 100000	cal o/p	0 099866	error	0 134081 %
given o/p	0 100000	cal o/p	0 099866	error	0 134081 %
given o/p	0 100000	cal o/p	0 099866	error	0.134081 %
given o/p	0 100000	cal o/p	0 102046	error	-2 046145 %
given o/p	0 100000	cal o/p	0.102046	error	-2.046145 %
given o/p	0 100000	cal o/p	0.102046	error	-2 046145 %
given o/p	0 100000	cal o/p	0.099926	error	0 074200 %
given o/p	0 100000	cal o/p	0 099926	error	0 074200 %
given o/p	0.100000	cal o/p	0 099926	error	0 074200 %
STEp	5				
given o/p	0 100000	cal o/p	0 100224	error	-0 224262 %
given o/p	0 100000	cal o/p	0 100224	error	-0 224262 %
given o/p	0 100000	cal o/p	0 100224	error	-0 224262 %
given o/p	0 100000	cal o/p	0 102388	error	-2.387807 %
given o/p	0 100000	cal o/p	0 102388	error	-2 387807 %
given o/p	0 100000	cal o/p	0 102388	error	-2 387807 %
given o/p	0 100000	cal o/p	0 100234	error	-0 234477 %
given o/p	0 100000	cal o/p	0.100234	error	-0 234477 %
given o/p	0 100000	cal o/p	0 100234	error	-0 234477 %

O/P = OUTPUT

RESULTS OF THIRD TRAINING SET

64 INPUT NODES, 1000 HIDDEN NODES, 3 OUTPUT NODES

3 PATTERNS, 1000 ITERATIONS.

```
               STEP NO             0
given o/p   0 100000    cal o/p   0 004103    error    95 896843 %
given o/p   0 100000    cal o/p   0 004103    error    95 896843 %
given o/p   0 100000    cal o/p   0 004103    error    95 896843 %
given o/p   0 100000    cal o/p   0 004119    error    95 881180 %
given o/p   0 100000    cal o/p   0 004119    error    95 881180 %
given o/p   0 100000    cal o/p   0 004119    error    95 881180 %
given o/p   0 100000    cal o/p   0 004171    error    95 829414 %
given o/p   0 100000    cal o/p   0 004171    error    95 829414 %
given o/p   0 100000    cal o/p   0 004171    error    95 829414 %
given o/p   0 100000    cal o/p   0 004125    error    95 874611 %
given o/p   0 100000    cal o/p   0 004125    error    95 874611 %
given o/p   0 100000    cal o/p   0 004125    error    95 874611 %
given o/p   0 100000    cal o/p   0 004220    error    95 780357 %
given o/p   0 100000    cal o/p   0 004220    error    95 780357 %
given o/p   0 100000    cal o/p   0 004220    error    95 780357 %
given o/p   0 100000    cal o/p   0 004104    error    95 895538 %
given o/p   0 100000    cal o/p   0 004104    error    95 895538 %
given o/p   0 100000    cal o/p   0 004104    error    95 895538 %
 STEp              1
given o/p   0 100000    cal o/p   0 100049    error    -0 048719 %
given o/p   0 100000    cal o/p   0 100049    error    -0 048719 %
given o/p   0 100000    cal o/p   0 100049    error    -0 048719 %
given o/p   0 100000    cal o/p   0 099886    error     0 114150 %
given o/p   0 100000    cal o/p   0 099886    error     0 114150 %
given o/p   0 100000    cal o/p   0 099886    error     0 114150 %
given o/p   0 100000    cal o/p   0 099831    error     0 168882 %
given o/p   0 100000    cal o/p   0 099831    error     0 168882 %
given o/p   0 100000    cal o/p   0 099831    error     0 168882 %
given o/p   0 100000    cal o/p   0 099951    error     0 048660 %
given o/p   0 100000    cal o/p   0 099951    error     0 048660 %
given o/p   0 100000    cal o/p   0 099951    error     0 048660 %
given o/p   0 100000    cal o/p   0 099748    error     0 251532 %
given o/p   0 100000    cal o/p   0 099748    error     0 251532 %
given o/p   0 100000    cal o/p   0 099748    error     0 251532 %
given o/p   0 100000    cal o/p   0 100111    error    -0 111438 %
given o/p   0 100000    cal o/p   0 100111    error    -0 111438 %
given o/p   0 100000    cal o/p   0 100111    error    -0 111438 %
 STEp              2
given o/p   0 100000    cal o/p   0 100229    error    -0 229411 %
given o/p   0 100000    cal o/p   0 100229    error    -0 229411 %
given o/p   0 100000    cal o/p   0 100229    error    -0 229411 %
given o/p   0 100000    cal o/p   0 100072    error    -0 071876 %
given o/p   0 100000    cal o/p   0 100072    error    -0 071876 %
given o/p   0 100000    cal o/p   0 100072    error    -0 071876 %
given o/p   0 100000    cal o/p   0 100003    error    -0 002742 %
given o/p   0 100000    cal o/p   0 100003    error    -0 002742 %
given o/p   0 100000    cal o/p   0 100003    error    -0 002742 %
given o/p   0 100000    cal o/p   0 100129    error    -0 129312 %
given o/p   0 100000    cal o/p   0 100129    error    -0 129312 %
given o/p   0 100000    cal o/p   0 100129    error    -0 129312 %
given o/p   0 100000    cal o/p   0 099933    error     0 067122 %
given o/p   0 100000    cal o/p   0 099933    error     0 067122 %
given o/p   0 100000    cal o/p   0 099933    error     0 067122 %
given o/p   0 100000    cal o/p   0 100276    error    -0 275634 %
given o/p   0 100000    cal o/p   0 100276    error    -0 275634 %
given o/p   0 100000    cal o/p   0 100276    error    -0 275634 %
 STEp              3
given o/p   0 100000    cal o/p   0 099756    error     0 243671 %
given o/p   0 100000    cal o/p   0 099756    error     0 243671 %
given o/p   0 100000    cal o/p   0 099756    error     0 243671 %
given o/p   0 100000    cal o/p   0 099600    error     0 399776 %
given o/p   0 100000    cal o/p   0 099600    error     0 399776 %
given o/p   0 100000    cal o/p   0 099600    error     0 399776 %
given o/p   0 100000    cal o/p   0 099530    error     0 470169 %
given o/p   0 100000    cal o/p   0 099530    error     0 470169 %
given o/p   0 100000    cal o/p   0 099530    error     0  70169 %
given o/p   0 100000    cal o/p   0 099656    error     0 344157 %
given o/p   0 100000    cal o/p   0 099656    error     0 344157 %
given o/p   0 100000    cal o/p   0 099656    error     0 344157 %
given o/p   0 100000    cal o/p   0 099472    error     0 528231 %
given o/p   0 100000    cal o/p   0 099472    error     0 528231 %
given o/p   0 100000    cal o/p   0 099472    error     0 528231 %
given o/p   0 100000    cal o/p   0 099790    error     0 209942 %
given o/p   0 100000    cal o/p   0 099790    error     0 209942 %
given o/p   0 100000    cal o/p   0 099790    error     0 209942 %
 STEp              4
given o/p   0 100000    cal o/p   0 100423    error    -0 422515 %
given o/p   0 100000    cal o/p   0 100423    error    -0 422515 %
given o/p   0 100000    cal o/p   0 100423    error    -0 422515 %
given o/p   0 100000    cal o/p   0 100264    error    -0 264212 %
given o/p   0 100000    cal o/p   0 100264    error    -0 264212 %
given o/p   0 100000    cal o/p   0 100264    error    -0 264212 %
given o/p   0 100000    cal o/p   0 100183    error    -0 182733 %
given o/p   0 100000    cal o/p   0 100183    error    -0 182733 %
given o/p   0 100000    cal o/p   0 100183    error    -0 182733 %
given o/p   0 100000    cal o/p   0 100317    error    -0 317045 %
given o/p   0 100000    cal o/p   0 100317    error    -0 317045 %
given o/p   0 100000    cal o/p   0 100317    error    -0 317045 %
given o/p   0 100000    cal o/p   0 100129    error    -0 128746 %
given o/p   0 100000    cal o/p   0 100129    error    -0 128746 %
given o/p   0 100000    cal o/p   0 100129    error    -0 128746 %
given o/p   0 100000    cal o/p   0 100438    error    -0 438213 %
given o/p   0 100000    cal o/p   0 100438    error    -0 438213 %
given o/p   0 100000    cal o/p   0 100438    error    -0 438213 %
 STEp              5
given o/p   0 100000    cal o/p   0 100242    error    -0 242427 %
given o/p   0 100000    cal o/p   0 100242    error    -0 242427 %
given o/p   0 100000    cal o/p   0 100242    error    -0 242427 %
given o/p   0 100000    cal o/p   0 100089    error    -0 089310 %
given o/p   0 100000    cal o/p   0 100089    error    -0 089310 %
given o/p   0 100000    cal o/p   0 100089    error    -0 089310 %
given o/p   0 100000    cal o/p   0 099997    error     0 002518 %
given o/p   0 100000    cal o/p   0 099997    error     0 002518 %
given o/p   0 100000    cal o/p   0 099997    error     0 002518 %
given o/p   0 100000    cal o/p   0 100135    error    -0 135362 %
given o/p   0 100000    cal o/p   0 100135    error    -0 135362 %
given o/p   0 100000    cal o/p   0 100135    error    -0 135362 %
given o/p   0 100000    cal o/p   0 099958    error     0 041582 %
given o/p   0 100000    cal o/p   0 099958    error     0 041582 %
given o/p   0 100000    cal o/p   0 099958    error     0 041582 %
given o/p   0 100000    cal o/p   0 100241    error    -0 240766 %
given o/p   0 100000    cal o/p   0 100241    error    -0 240766 %
given o/p   0 100000    cal o/p   0 100241    error    -0 240766 %
```

O/P = OUTPUT

RESULTS OF FOURTH TRAINING SET

84 INPUT NODES, 500 HIDDEN NODES, 3 OUTPUT NODES

8 PATTERNS, 500 ITERATIONS

CHAPTER 6

SUMMARY,CONCLUSION AND SCOPE FOR FUTURE WORK

6.1 SUMMARY

There are two main objectives of the present study :

1. To have a deep insight into the field of ANNs (which are complementry to Artificial Intelligence) gaining sufficient knowledge about the principles involved therein and their working and also to study in depth the phenomena of Image Processing and Analog to Digital Conversion. Not only this, the purpose is to examine how the artificial neural networks can be used in association with a good Vision System to locate and identify 3D objects in the environment autonomously using a single and arbitrary view . Thus a knowledge of the present *state of this art of 3D visual scene analysis* using ANN has to be gained so as to apply this,in principle, to the field of (image processing in) traffic engineering.
2. To attempt shape recognition of different types of vehicles seen in a 3D visual scene by training an ANN for this purpose.

A detailed study has been conducted to fulfill the first objective and the details have been noted down in chapters 3

- -

In order to fullfill the second objective, 2D pictures frames of the front views of the vehicles have been digitized (converted into numbers, the laguage of computers) as per a definite scheme (for details refer chap. 5). The digitized inputs have to be kept in the form of low order matrices because for high order matrices (256x256 or 512x512) memory as high as 100MB or even more is required which posed a serious restriction and must have affected the results adversely. Even with small matrices of the order of 8x8 (i.e 64 input nodes), the no. of hidden layer nodes couldn't be increased beyond a certain value due to the memory restriction. A software using Artificial Neural Network theory has been developed wherein the algorithm used for training thr network is the Backpropagation algorithm. This network was trained with different input patterns (digitized inputs) and it has been found that the error in a good number of cases gets reduced to an acceptably low value which is a very encouraging result. The results of training for different patterns and after varying the parameters affecting convergence have been shown in the Chapter 5. However, some cases also give a high percentage of error and convergence is not achieved even after a lot many iterations which may be due to :

(i) low order of input matrices which has been kept due to

limitation of memory space.

(ii) caveats of BP algorithm like local minima etc.

(iii) network bias against such input patterns (refer article 3.4)

6.2 CONCLUSIONS

The successful training of the network by a number of classes of patterns, even with memory restriction, is a healthy sign towards future use of ANNs for image processing but it would be better to use special hardware for ANNs if better results are to be reached at because the large decrease in the order of input matrices owing to the memory restriction must have affected the results adversely.

The network has a bias towards certain types of patterns as compared to other types which supports and is analogous to *Darwin's Theory of Natural Selection* [refer Article 3.4 for details].

The time consumed for training is very large and an attempt can be made to make use of Genetic Algoritms to reduce the same.

It can be concluded that neural networks are potentially of considerable value in the field of traffic engineering but a major problem in their use is the acquisition of large enough data sets for effective training and the high memory requirement during training besides the large CPU time needed

for training.

6.3 SCOPE FOR FUTURE WORK

The present work may be taken to mark an effective begining in the use of ANNs in association with Vision systems for 3D image processing in trafic engineering.There is a large scope for following future work in this field of automatized image procesing for visual scene analysis:

1. To work out dynamics of vehicles.
2. Designing of automatized intersection control.
3. Design of automatized accident warning systems; such a device may be fixed at the front and rear of each vehicle so as to give a warning signal, whenever the vehicle intrudes in the headway distance, as per relative speed of following/leading vehicle.
4. Proper traffic monitoring at congested links
5. Proper maintenance and repair of pavements.
6. Study of driver behaviour and using the results for future studies.
7. Counting of vehicles with a particular type of number plates thus replacing the manual counting.
8. Detection of defects in vehicles which can save much time in comparison to manual detection.
9. Automated vehicle dispatching with efficiency compared to that of an expert dispatcher.

REFERENCES

Amari, S. (1983), "Field Theory of Self Organizing Neural Nets" , IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, pp 741-748.

Barto, A.G., and Anandan, P. (1985), "Pattern Recognizing Stochastic Learning Automata", IEEE Transactions on Systems, Man, and Cybernetics, 15, pp 360-375.

Cheu, R.L., Ritchie, S.G., Recker, W.W., and Bavarian, B. , "Investigations of a neural network model for freeway incidence detection", Irvine Institute of Transportation Studies, University of California, Irvine.

Choudhary, S.D., (1992), "Development of a Fast Neural Network Simulator with Specific Application to Electric Load Forecasting Problem", B.Tech. Project Report, I.I.T. Kanpur, India.

Dickension, K.W., and Wan, C.L., "Road traffic monitoring using the TRIP 2 system",Institution of Electrical Engineers, London.

Duda, R.O., Hart, P.E. (1973), Pattern Classification and Scene Analysis, New York, Wiley.

Enquist, M., and Arak, A. (Feb. 1993), "Selection of exaggerated male traits by female aesthetic sense", Nature,Vol. 361.

Hwang, K., and DeGroot,D. (1989), Parallel Processing for Super Computers and Artificial Intelligence, Mc Graw Hill Pubishing Company, Singapore.

Hillier, F.S., and Lieberman, G.J. (1973), Operations Research, Holden Day Publications.

Kaseko, M.S., and Ritchie, S.G., "Pavement image processing using neural networks", Irvine Institute of Tranportation Studies, University of California, Irvine.

Kirby, H., Boyle, R., and Dougherty, M. (1993), "The use of neural networks to recognise and predict traffic congestion".

Kohonen et al. (1991), Artificial Neural Networks, Volume 1., Elsevier Publishing Company B. V., Amsterdam, Netherlands.

Lippmann, R.P., and Gold, B. (1987), "Neural Net Classifiers Useful for Speech Recognition", Proceedings of the First International Conference on Neural Networks, San Diego, IEEE Computer Society Press.

Margarita, G. (1990), "Recognition of European car plates with modular neural network", International Neural Network Conference Report, pp 408 - 411.

Mc Clelland, J.L., Rumelhart, D.E., and PDP Research Group (1986). Parallel Distributed Processing, Volume 1 and 2, MIT Press, Cambridge.

Nakasuji, T., and Kaku,T. (1991), "Development of a Self Organizing Traffic Control System using Neural Network Models" ,Tranportation Research Record No. 1324, pp 137 - 145, Transportation Research Board, Washington D.C., USA.

Nilson, N. (1965), Learning Machines, McGraw Hill, New York.

Patterson, D.W. (1990), Artificial Intelligence and Expert Systems, Prentice Hall of India Private Limited, New Delhi.

Pomerleau, D.A., "ALVINN- an autonomous land vehicle in a

neural network" ;Technical Report No. 77, Computer Science Department, Carnarnegie Mellon University.

Ritchie, S.G. , Kaseko, M. , and Bavarian, B. (1991) , "Development of an intelligent system for automated pavement evaluation" , Tranportation Research Record No. 1311, pp 112 - 119, Tranportation Research Board, Washington D.C., USA

Rumelhart, D.E., Mc Clelland, J.L. (1987), Explorations in Parallel Distributed Processing - A Handbook of Models Programs and Exercises, MIT Press, Cambridge, USA.

Sabbah, D. (1985), "Computing with Connections in Visual Recognition of Origami Objects", Cognitive Sciences, 9, pp 25-50.

Sandon, P.A. and Uhr, L.M. (1988), "A Local Interaction Heuristic for Adaptive Networks", Proceedings of the International Conference on Neural Networks San Diego IEEE Computer Sociey Press.

Simpson, P.K., (1989). Artificial Neural Systems - Foundations, Paradigms, Applications and Implementations, Pargamon Press Inc., Elmsford, New York.

Takubo, N. (1991), "A study of driver's behaviour using a driving simulator", Kotsuhen Report Volume 32, No.1 pp 43 -54.

Touretzsky, D.S. , and Hinton, G.E. (1985), "Symbols Among the Neuron: Details of a Connectionist Inference Architecture", Proceedings on the International Joint Conference on Artificial Intelligence", Los Angles.

Wasserman, Philip de, (1989), Neural Computing, Van Nostrand Reinhold, 115, Fifth Avenue, New York.

Widrow, B., and Stearns, S.D. (1985), Adaptive Signal Processing, Prentice Hall.